TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

TUOMAS NIEMINEN

DEEP LEARNING IN QUANTIFYING VASCULAR BURDEN FROM BRAIN IMAGES

Master of Science thesis

Examiner: Prof. Hannu Eskola
Examiner and topic approved by the
Faculty Council of the Faculty of
Computing and Electrical Engineering
on 28th February 2018

ABSTRACT

TUOMAS NIEMINEN: Deep learning in quantifying vascular burden from brain images
Tampere University of Technology
Master of Science thesis, 57 pages
April 2018
Master's Degree Programme in Electrical Engineering
Major: Biomedical Engineering
Examiner: Prof. Hannu Eskola
Keywords: white matter hyperintensities, brain infarcts, deep learning, convolutional neural networks, segmentation

White matter hyperintensities (WMH) and lacunar infarcts are features of cerebral vessel disease. Together with cortical infarcts they are main causes of vascular dementia. Also, increased WMH volume is associated with the risk of Alzheimer's disease. This is the reason why an accurate automatic WMH and infarct segmentation tool is highly desirable in order to improve dementia diagnosis.

In this thesis deep learning, more precisely, convolutional neural network called uResNet was used to segment WMH, lacunar infarcts and cortical infarcts from brain images. The study was done by training the network using multiple different input channel sets. Also, the amount of classes to be segmented varied. In total 21 different combinations were trained and tested including both 2D and 3D models.

The numerical and visual evaluation was performed by comparing result images to the expert annotaded images. Numerical evaluation included computation of Dice scores and correlation between the image sets. Also, for infarct detection sensitivities and amount of false positive segmentations were calculated. From the results can be deduced that proposed segmentation method is capable of accurate WMH segmentation (best achieved Dice score for WMH volumes was 0.774). However, further research is still needed in order to improve infarct segmentation results since sensitivity scores were surprisingly poor and the amount of false positive segmentations was high.

TIIVISTELMÄ

TUOMAS NIEMINEN: Verenkiertohäiriöiden kvantifiointi aivokuvista syväoppimismenetelmällä
Tampereen teknillinen yliopisto
Diplomityö, 57 sivua
Huhtikuu 2018
Sähkötekniikan koulutusohjelma
Pääine: Biolääketieteen tekniikka
Tarkastaja: Prof. Hannu Eskola
Avainsanat: valkean aineen hyperintensiteetit, aivoinfarktit, syväoppiminen, kovoluutioneuroverkot, segmentointi

Aivojen valkean aineen hyperintentiseetit (WMH) ja infarktit ovat aivojen verenkiertohäiriöitä, jotka aiheuttavat vaskulaarista dementiaa. Kohonnut WMH-tilavuus viittaa myös Alzheimerin tautiin. Tämän vuoksi automaattinen työkalu, joka osaa segmentoida tarkasti WMH:n lisäksi myös aivoinfarktit omiksi luokikseen aivokuvista olisi hyödyllinen dementian diagnosoinnissa.

Tämän diplomityön tavoitteena oli toteuttaa automaattinen segmentointimenetelmä uResNet kovoluutioneuroverkkoa hyödyntämällä aivojen WMH:lle ja aivoinfarkteille. Neuroverkko opetettiin usealla erilaisella opetustdatayhdistelmällä, joissa käytettyjen eri kuvantamissekvensseillä kuvattujen magneettikuvien ja segmentoitavien luokkien määrä vaihteli. Yhteensä tässä tutkimuksessa opetettiin ja testattiin 21 erilaista ydistelmää opetusdatan ollessa sekä kaksi- että kolmiulotteista.

Segmentointimenetelmän toimivuutta testattiin vertaamalla tuloskuvia vastaaviin ammattilaisen segmentoimiin kuviin laskemalla kuvasettien välisiä Dice-arvoja ja korrelaatioita. Lisäksi sensitiivisyydet ja väärien positiivisten segmentointien lukumäärä laskettiin malleille, jotka segmentoivat WMH:n lisäksi myös aivoinfarkteja. Numeeristen tulosten ja visuaalisen tarkastelun perusteella tässä työssä käytetty segmentointimenetelmä kykenee hyvään tarkkuuteen WMH:n segmentoinnissa (paras saavutettu Dice arvo oli 0.774 WMH tilavuuksille). Infarktien segmentointi kuitenkin vaatii vielä lisätutkimusta, sillä saavutetut sensitiivisyydet eivät vastanneet odotuksia ja väärien positiivisten segmentointien määrä oli liian suuri.

PREFACE

This thesis was done for Combinostics Oy between September 2017 and April 2018.

I want to thank my supervisor Juha Koikkalainen for all the advises and support during this work. Also, I'd like to thank the whole Combinostics team for the help and valuable insight.

Tampere, 20.04.2018

Tuomas Nieminen

CONTENTS

LIST OF ABBREVIATIONS AND SYMBOLS

AD	Alzheimer's disease
ANN	Artificial neural network
CNN	Convolutional neural network
CP	Control points
CPU	Central processing unit
CT	Computed tomography
CVD	Cerebral vessel disease
DOF	Degrees of freedom
EM	The Expectation–Maximization algorithm
FLAIR	Fluid-attenuated inversion recovery
FN	False negative
FP	False positive
GMH	Gray matter hyperintensities
GPU	Graphics processing unit
LADIS	Leukoaraiosis and Disability study
LPA	Lesion prediction algorithm
MI	Mutual information
MLP	Multilayer perceptron
MRI	Magnetic resonance imaging
MS	Multiple sclerosis
NMI	Normalized mutual information
PET	Positron emission tomography
R	Pearson correlation coefficient
ReLU	Rectified linear unit
RNN	Recurrent neural network
SGD	Stochastic gradient descent
SPECT	Single photon emission computed tomography
TE	Time echo
TR	Repetition time
VaD	Vascular dementia
WM	White matter
WMH	White matter hyperintensities
2D	2-dimensional
3D	3-dimensional
h	hours

m	meter
min	minutes
mm	millimeter
mm^3	cubic millimeter
ms	millisecond
s	seconds
T	Tesla

1. INTRODUCTION

Up to this day medical image interpretation in clinics and hospitals has mostly been performed by human expert such as physicists or radiologist and computer image analysis has been one step behind compared to advances in medical imaging technologies. Because of the possible human made error and huge number of different pathologies, computer aided interventions have been recently studied. Advances in machine learning, especially, in the field of deep learning have improved the ability to classify, quantify and identify patterns in medical images. [53] Deep learning is a broader term describing machine learning methods which consist of multiple data processing layers. These methods are based on learning the data representations. [34]

Deep learning methods, in particular convolutional neural networks (CNNs), have become the state-of-the-art methods for medical image analysis tasks due to fact that modern central processing units (CPUs) and graphics processing units (GPUs) are powerful enough to process huge amount of data with advanced learning algorithms.[37] Applying machine learning methods in image analysis has involved feature extraction which has usually been done by humans based on their expertise. This step is still done by human experts, but deep learning absorbs the feature engineering step into training step of the deep learning model, letting the computer learn features based on a set of preprocessed data. These deep learning models transform the input data such as images to outputs while learning the features. This makes it easy for non-experts to use deep learning methods and algorithms.

CNNs are applied in many image processing tasks such as image segmentation [38] or image classification [32]. Recently CNNs are also applied to medical image processing [27] [36]. In medical imaging the data comes from a variety of imaging technologies such as magnetic resonance imaging (MRI), computed tomography (CT) or positron emission tomography (PET). Usually data is 2D or 3D describing different anatomical structures such as bones, major organs or brain tissue along with possible unhealthy structures such as bone fractures, tumors or lesions. Segmentation aims to outline different anatomical structures and detect unhealthy tissues. [29]

In this thesis we focus on segmenting white matter hyperintensities (WMH) and infarcts from brain images. Both WMH and infarcts are usually found from older patients with dementia [62]. WMH are small vessel disease and can be easily detected from fluid-attenuated inversion recovery (FLAIR) magnetic resonance images as a bright hyperintense regions. In addition to WHM regions, cortical infarcts can also appear as a hyperintense regions and lacunar infarcts as hypointense regions in FLAIR images. [47] WMH and lacunar infarcts are features of cerebral vessel disease and together with cortical infarcts they are main causes of vascular dementia. Also, increased WMH volume is associated with the risk of Alzheimer's disease [45]. Therefore, detecting and segmenting WMH, lacunar infarcts and cortical infarcts from brain images is clinically important.

In some studies WMH lesions and infarcts have been determined by hand and sometimes with the help on semi-automatic tool. This, however, is very time consuming for bigger datasets and an accurate automatic segmentation tool is highly desirable. [27] Therefore, the aim of this thesis is to develop an accurate segmentation method based on convolutional neural networks for detecting lesions related to vascular burden from MR images. The thesis is structured as follows. Chapter 2 is introducing dementia and vascular burden. Chapter 3 focuses on computer image analysis introducing the key theory behind the applied methods. Then chapter 4 presents the used materials, image analysis pipeline, image preprocessing and neural network structure. Chapter 5 will present the study results and it is followed by a discussion and conclusions chapters.

1.1 Related work

Multiple automated and semi-automated segmentation tools have been presented over the years for WMH and infarct segmentation. Those methods can be divided into supervised and semi-supervised methods. Supervised methods are using predefined training data annotated by human expert as a "ground truth". When only a fraction of the training data is labeled, method is called semi-supervised learning and unsupervised learning when no labeled training data is available. [27].

Unsupervised methods extracting WMH regions from brain images are not widely used but few methods exists. Jack et al. [24] segmented WMH by using a simple threshold derived from a regression analysis on the histogram of the FLAIR images. More robust way statistically threshold WMH is to derive white matter (WM) intensities from probabilistic atlas and based on the information received from T1 images, remove false positives [63]. More recently, Erihov et al. [11] proposed a method that exploits brain asymmetry which is a saliency-based method. Also, other

methods relying on random forests and Gaussian mixture models have been proposed [65] [7]. One example for method based on Gaussian mixture models is proposed by Wang et al.[61] in which Gaussian mixture models are used to model FLAIR intensity distribution and then Expectation–Maximization (EM) algorithm estimates the intensity mean and standard deviation for each tissue class. However, most of these techniques work better for lesion detection instead of segmentation.

For semi-supervised segmentation several semi-automatic segmentation algorithms exists. These algorithms rely mostly on region growing algorithms where number of seed points are initialized manually [16]. Region growing algorithms are also semi-automatic models. One semi-supervised segmentation algorithm is proposed by Qin et al. [44] in which the idea is to maximize the margin over the inliers and outliers. Other method extracts WMH with region growing by spreading seed points into neighborhood where intensity values are bigger than the selected threshold value [23]. However, semi-supervised WMH segmentation methods are not performing well compared to supervised segmentation methods.

Supervised WMH segmentation methods are based on many different models and algorithms such as random forests, logistic regression models, support vector machines and neural networks, especially, convolutional neural networks [16]. Logistic regression model, known as lesion prediction algorithm (LPA), is trained with the data from 53 multiple sclerosis (MS) patients [52]. The data consisting of binary lesion maps of these 53 patients serve as a response values and different lesion maps that take voxel specific changes into account were used as spatial covariates. The problem with most of the previous methods are that they are not very good at handling data with multiple classes and this is the reason why the best performing WMH segmentation methods are currently based on convolutional neural networks which are able to model complicated non-linear functions needed in WMH segmentation tasks [16]. There are many different convolutional neural networks proposed for WMH segmentation tasks such as Ronneberger et al. [46] U-shaped network architecture or Kammnitsas et al. [27] multi-channel multi-resolution 3D CNN, which uses a different input channel for each resolution and then deeper merges them in order to produce a prediction.

2. DEMENTIA AND VASCULAR BURDEN

In this chapter the most common types of dementia and vascular burden are introduced.

2.1 Dementia

Dementia is a group of brain diseases that affect person's ability to think and remember beyond what is expected from normal aging. It causes problems in brain functions and most common symptoms are for example problems with thinking, memory, learning capacity and communication. However, consciousness is usually not affected. Dementia can be caused by many diseases and injuries that affect the brain such as infarcts or Alzheimer's disease (AD). Dementia is one of major causes of disability and dependency for older people. It is estimated that 5 to 8 people out of the population of 100 people who are aged over 60 years are suffering from dementia. Other common causes of dementia are vascular dementia (VaD), Parkinson's disease, Huntington's disease and Creutzfeldt-Jakob disease. [64]

Alzheimer's disease is a chronic neurodegenerative disease and the most common cause of dementia. It is a very specific form of dementia and it usually starts slowly and worsens over the time. The most noticeable changes caused by AD in behavior are difficulty in learning, short-term memory loss, mood swings and problems with language. Slowly bodily functions are lost which can lead to death. The progress of the disease varies but the average life expectancy is three to five years after diagnosis. [66] The cause of AD is not understood properly, and it is believed that AD is caused by a combination of genetic, lifestyle and environmental factors that affect the brain over time. AD's neurological characteristics consist of neurofibrillary tangles, neuritic plaques and neuronal loss [9]. Focus on neuropathology is in the medial temporal lobe regions such as hippocampus, entorhinal cortex and subiculum. Alzheimer disease diagnosis is based on the cognitive testing with medical imaging along with history of the illness. [49]

Vascular dementia usually consists of any type of dementia caused by cerebral blood vessel diseases (CVD) which consist of pathological process of subcortical structures.

CVD usually contains lacunar infarcts, white matter lesions, Binwanger's disease and cerebral microbleeds but it is also responsible for cerebral infarcts, ischemic infarcts and encephalopathy. [5] For example, cognitive decline can be caused by a series of brain infarcts due to problems in large vessels or due to changes in small vessels of the white matter. Basically, VaD is an umbrella term for group of lesions which block or disturb blood flow in brain.[15]

2.2 Imaging in vascular dementia

Advanced medical imaging technologies such as CT, MRI, positron emission tomography (PET) and single-photon emission computed tomography (SPECT) are the most common imaging techniques to study white matter changes, cerebral pathologies or subtypes of dementia in older persons. [45] The focus on this thesis is to study white matter changes and brain infarcts using FLAIR, T1 and T2 MR images.

2.2.1 White matter hyperintensities

White matter lesions, also known as leukoaraiosis, can be visualized as hyperintensities on FLAIR and T2 MR images. [45] In T1 images white matter has high signal intensity and white matter lesions appear as lower intensity regions. [50] White matter lesions most commonly reflect to CVD and these regions of high intensities in FLAIR and T2 images can be found within cerebral white matter and are called as WMH. Hyperintensities can be found also within subcortical gray matter and then they are referred as gray matter hyperintensities (GMH). WMH are common for older people but they are also seen in several neurological disorders and illnesses. Especially, increased WMH volume is associated with the progression and risk of Alzheimer disease. WMH are caused by many different factors such as ischemia, gliosis or breaches of the barrier between the brain and cerebrospinal fluid. [45] WMH are visualized in Figure 2.1.

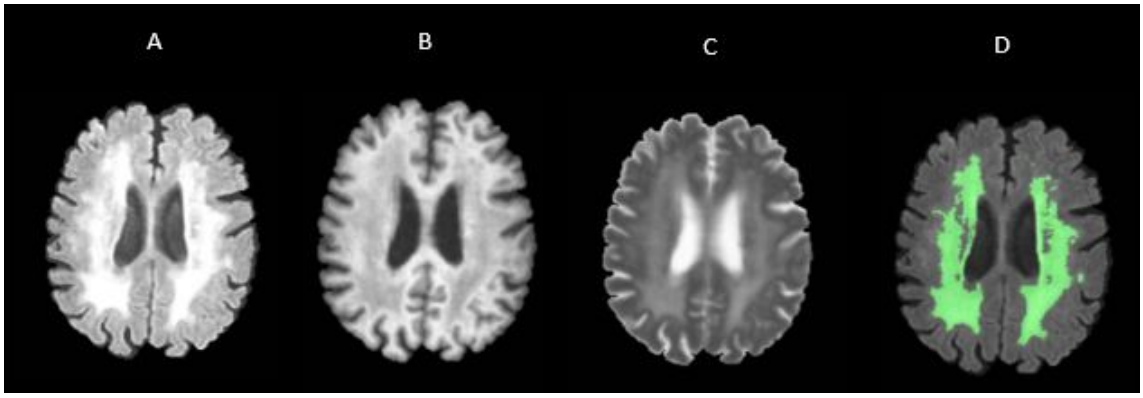

Figure 2.1 *Different MR images with WMH. Non-brain tissues and structures are removed from the images. FLAIR (A), T1 (B), T2 (C) and FLAIR image with highlighted WMH lesions (D).*

2.2.2 Infarcts

Condition when blood flow is bad in the brain causing cell death is called as an infarct and infarcts can be divided into two groups. Hemorrhagic infarcts are caused by bleeding directly in brain or in the space between the brain's membranes. Infarcts caused by lack of blood flow often due to blockage of a blood vessel are called as ischemic infarcts. Both of them result in part of the brain not functioning properly and the most common symptoms include inability to move or feel, loss of vision, problems with understanding and speaking. Infarct may affect different cortical regions of the cerebral cortex and this spatial differentiation is clinically important. The etiology and clinical management for cortical and subcortical infarct may differ and cortical infarcts can affect higher cognitive functions depending on the side of the brain and the lobe involved. For example, a tiny infarct is usually due to a blockage of small penetrating artery, whereas a middle cerebral artery occlusion resulting in a cortical infarct usually results from an embolus from either the heart, aortic arch or carotid artery. [10] [8] In FLAIR and T2 MR images acute cortical infarcts appear as large hyperintense regions in cerebral cortex. In T1 images cortical infarcts can be seen as low intensities. If infarct is in chronic state, infarct can be seen as low intensities in FLAIR and T2 images in those areas where brain tissue has died. [20]

Small infarcts in the distal distribution of deep penetrating vessels are called lacunar infarcts. Their diameter is smaller than 20 mm and they result from occlusion of single penetrating artery at the base of the brain. Lacunar infarcts appear when one of the penetrating arteries that provide blood to the brain's deep distributions is blocked. Lacunar infarct can affect cognitive and motor functions such as sight,

movement, coordination or speech because these functions are controlled by different areas of the brain in which lacunar infarct may onset. Lacunar infarct diagnosis is mainly done according to the neuroimaging and clinical studies. Lacunar infarcts cause impairment to the cells in small parts of the brain leading to, in a worst case, death of the brain tissue. However, in many cases there are not any outside symptoms, but lacunar infarct still destroys little by little brain functions and increases the risk of major infarct. Occurrence of multiple lacunar infarcts, aging and role of risk factors can finally lead to dementia. Also, AD and lacunar infarcts have group of overlapping risk factors such as hypertension, diabetes, hyperlipidemia and unhealthy lifestyle but the relationship between the AD and lacunar infarct is still unclear.[5]

Lacunar infarcts appear in FLAIR images as small hypointense regions with surrounding hyperintense rim. However, sometimes rim is not present. Also in some cases, the cavity of the lacunar infarct is not visible in FLAIR images and the lacunar infarct appears entirely hyperintense. In T1 images lacunar infarcts can be seen as hypointense regions without the rim and in T2 images they are hyperintense regions. [12] In the Figure 2.2 cortical and lacunar infarcts are visualized alongside with WMH.

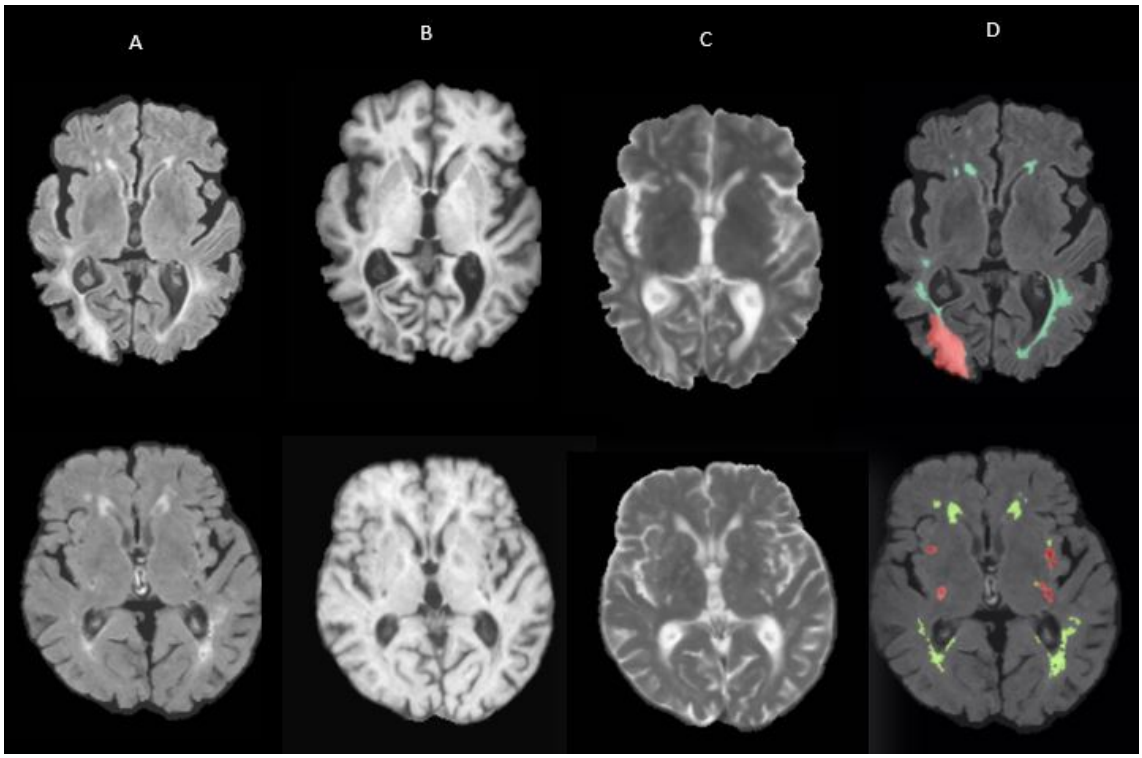

Figure 2.2 *Different MR images with WMH and either cortical infarct (top row) or lacunar infarcts (bottom row). Non-brain tissues and structures are removed from the images. FLAIR (A), T1 (B), T2 (C) and FLAIR image with highlighted WMH and infarct lesions (D). Red areas are infarct tissue and green areas WMH.*

3. COMPUTER IMAGE ANALYSIS

This part of the thesis describes modern image processing techniques and focus is on those techniques and models that are important in this thesis. In addition to image processing techniques, neural networks and deep learning are discussed since they are becoming more and more important in modern image analysis. Image analysis pipeline containing deep learning is implemented later in this thesis.

3.1 Image registration

In image registration two or more images are combined in order to provide complementary information. For example, images that represent the same scene taken from different viewpoints or by different imaging sequence are transformed into a same coordinate system. In other words, registration aligns these images geometrically in order to gain information from various combined data sources. Registration is commonly used in many applications such as computer vision, multispectral classification, change detection, cartography, environmental monitoring, medicine and many more. In medicine registration plays a crucial role in combining CT or MRI data to obtain more accurate information about the patient, monitor tumor growth and verificate treatments among others. [67]

Usually image registration requires selection of the feature space, a similarity measure and search strategy. There are many different registration methodologies presented and they can be classified using the feature space image information which can be related to the image voxel intensities, intensity gradients, statistical information of the voxel intensities or extracted image features. Methods based on image voxel intensities are known as intensity based methods and methods based on extracted image features as feature-based methods. There are also other methods which are based on image frequency domain or Fourier transform. [42]

In the Figure 3.1 intensity based registration process is visualized. The registration process searches iteratively a geometric transformation which optimizes the similarity measure. In this case, similarity measure, also known as loss function, is related to the image voxel intensities and aim is to minimize or maximize it. The optimizer

defines the search strategy and interpolator resamples the voxel intensities into the new coordinate system based on found geometric transformation. [42]

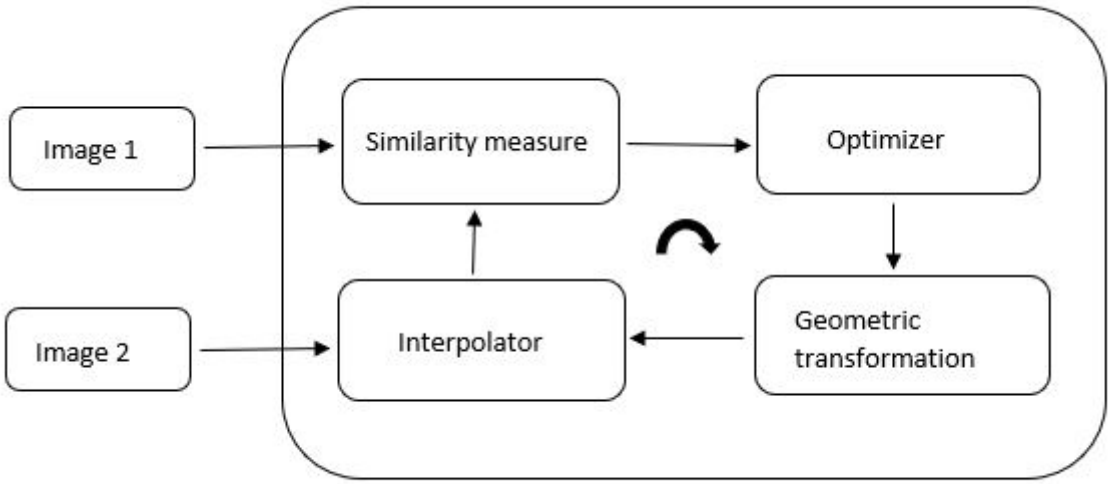

Figure 3.1 *Intensity based image registration process. Modified from [42].*

Feature-based registration methods, on the other hand, are based on extracting structures from the images. Typical feature-based registration process is visualized in Figure 3.2 and from the figure can be seen that feature-based methods usually consist of four steps. First step is feature detection in which objects such as edges, contours or line intersections are manually or automatically detected from the images and represented by their point representatives. However, medical images tend to not have enough clear details and area-based methods, which focus more on feature matching step leaving feature detection step out, are usually performing better. Second step is feature matching which detects the correspondence between the features in source and target images using many similarity measures and feature descriptors. After feature matching, mapping function is constructed based on the correspondence estimated in the previous step. Finally, image resampling transforms the image into desired coordinate system based on the constructed mapping function. [67]

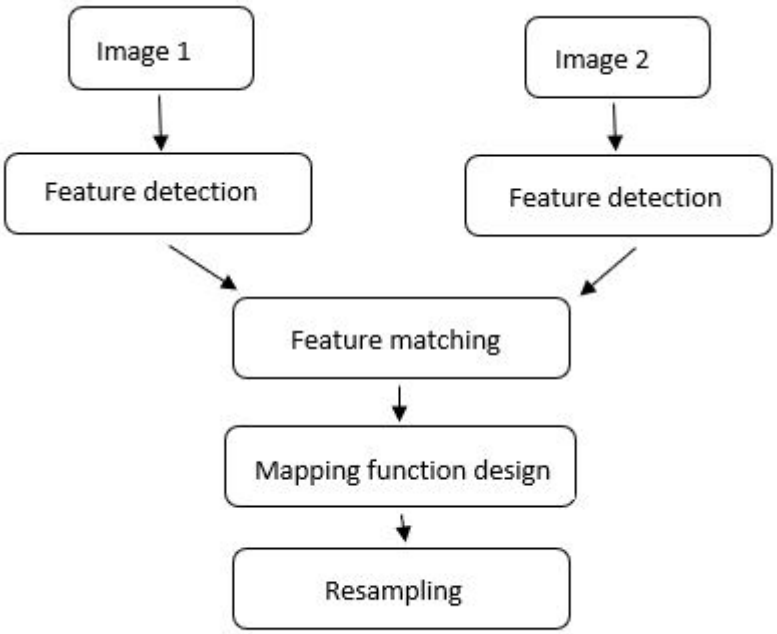

Figure 3.2 *Feature-based image registration process. Modified from [42].*

3.1.1 Similarity measures

Similarity measures are playing crucial role in medical image registration and one of the simplest one is sum of squared differences (SSD) which is defined as

$$SSD = \frac{1}{N} \sum_{x_A \epsilon \Omega} |A(x) - B^{\tau}(x)|^2, \tag{3.1}$$

where A and B^{τ} are different images and Ω is image domain. However, SSD measure assumes that after registration, the images differ only by Gaussian noise. Therefore normalized cross-correlation (CC) could be better choice. CC is defined as

$$CC = \frac{\sum_{x_A \epsilon \Omega}(A(x_A) - \bar{A})(B^{\tau}(x_A) - \bar{B})}{\sqrt{\sum_{x_A \epsilon \Omega}(A(x_A) - \bar{A})^2 \sum_{x_A \epsilon \Omega}(B^{\tau}(x_A)) - \bar{B})^2}}, \tag{3.2}$$

where $\bar{A}$ is the mean voxel value in image A and $\bar{B}$ is the mean of B^{τ}. [21] The major drawbacks for cross-correlation are high computational complexity and flatness of the similarity maxima but it is still widely used because of easy hardware implementation. [67]

A leading similarity measure in multimodal image registration is mutual information (MI). It measures the statistical dependency between the two different data sets and in registration the goal is to maximize it. Mutual information between two random variables X and Y is defined as

$$MI(X,Y) = H(X) + H(Y) - H(X,Y), \tag{3.3}$$

where $H(X)$ and $H(Y)$ represent the entropy of a random variables and is defined as

$$H(X) = -E_x(log(P(X))), \tag{3.4}$$

where P(X) is the probability distribution of X. [67] Sometimes changes in overlap of the low-intensity regions of the image affect MI too much and in order to overcome this problem normalized mutual information (NMI) is used. NMI is defined as

$$NMI(X,Y) = \frac{H(X) + H(Y)}{H(X,Y)}.[21] \tag{3.5}$$

In image registration the most challenging tasks are registration of images with complex nonlinear and local distortions, multimodal registration and registration of multidimensional images. In multimodal registration MI is the mainly used method especially in medical image registration but it also has some limitations especially when images have high rotation and scaling differences. [67]

3.2 Image segmentation

Image segmentation is one of the most critical tasks in medical image analysis because many medical applications require detecting specific regions from the images such as tissues or organs. Medical images contain a lot of information and often only one or two structures are interesting. Image segmentation is a tool for extracting that interesting information from the images in order to help medical experts in diagnostics, planning and guidance. Image segmentation refers to process where a digital image is partitioned into multiple segments consisting a set of pixels. Basically, segmentation changes the representation of an image so that specific regions or objects are easier to detect and analyze. Usually segmentation methods segment some objects or boundaries from the images and this is done by assigning a label

for every pixel. Then pixels with the same label are belonging in the same segment. Resulting image is a set of different image segments that cover the entire original image. Result of the image segmentation process can be seen in the Figure 3.3. In the figure white matter tissue, gray matter tissue and cerebrospinal fluid are segmented from T1-weighted MR image. [54]

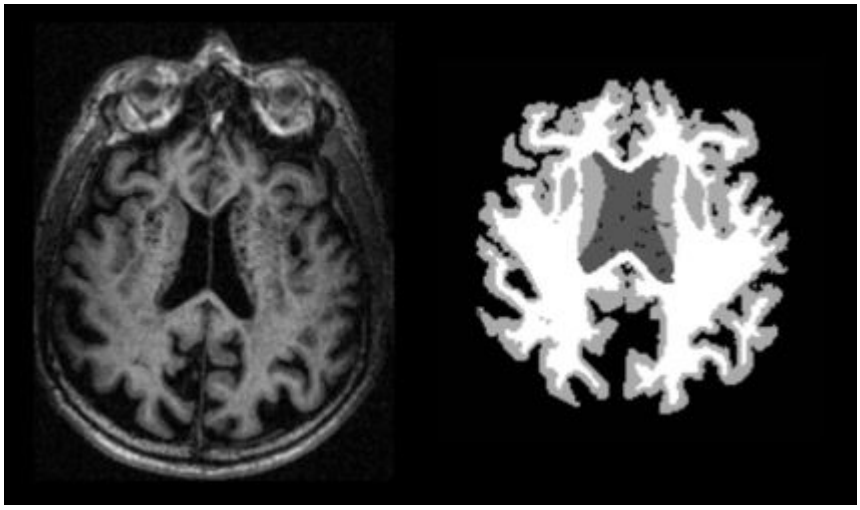

Figure 3.3 *White matter, gray matter and cerebrospinal fluid segmented from T1-weighted MR image. On the left-hand side is original T1-weighted image and on the right-hand side is tissue segmentation image.*

Image segmentation algorithms can be divided into several different categories and some popular segmentation methods can be found from the Table 3.1. First group consists of segmentation algorithms based on thresholding where grayscale images are converted into binary images by selecting an optimum threshold value. This binary image should contain all the necessary information about the region of interest. [54] Otsu's method is a thresholding method which figures out the optimal threshold that minimizes the weighted within-class variance in an image. It is suitable for converting grayscale images in to a binary images automatically. Gaussian mixture methods are also based on thresholding. They estimate the number of components with their means and covariance automatically using EM algorithm. [35]

Edge based segmentation methods, on the other hand, are based on finding the edges or boundaries of the object of interest. Edges and boundaries can be seen in the images as discontinuities within image intensities. [54] Edge detection methods are important for object recognition of human organs in medical images and one popular edge based method is watershed. [35] In watershed different gradient values are considered as different heights and from each local minimum a water will raise towards the local maximum. When two body of water meet, a dam is built between them and regions are separated. [2]

Third group of segmentation methods consist of region-based methods in which seed points are initialized in the middle of an object and then algorithm grows the labeled area until it meets the object boundaries. [54] For example, region growing method proposed by Adams [1] expands the seed region by merging unallocated neighbor pixels which have the smallest difference between the region and the pixel.

Clustering is also one way to perform segmentation. These techniques group similar patterns together, in other words, it determines which components of the data set are belonging in the same cluster. [54] Fuzzy c-means is a clustering algorithm which is widely used in medical image segmentation. It is based on minimizing the object function defined as

$$J_q = \sum_{i=1}^{n} \sum_{j=1}^{m} u_{ij}^q d(x_i, \Theta_j), \tag{3.6}$$

where q controls the fuzziness degree of clustering, u is fuzzy membership function of data x_i to cluster with center Θ_j and d is distance between center of the cluster j and data x_i. The aim is to optimize the object function by updating the membership function and centers of clusters until optimization between iterations are more than predefined threshold. [2] K-means is other clustering algorithm which partitions the data into k clusters. [35]

In addition to traditional image segmentation methods, more segmentation methods exists such as atlas-based segmentation methods or neural network -based segmentation methods. Atlases are used for segmentation when there is not enough contrast between the tissues in an MR image of the brain. Atlases are images which describe the common anatomy of the brain. In atlas-based segmentation, image is registered to the atlas which is used as a prior information in the image segmentation process.[2]

Neural networks are becoming more and more popular especially in medical image processing. [4] These deep learning methods are able to classify each pixel in the image individually based on huge amount of training data making it very fast method once the model is trained. [53] Rest of the thesis focus on applying deep learning methods in brain image segmentation tasks.

Name	Category
Grayscale thresholding	Thresholding-based
Otsu's method	Thresholding-based
Gaussian mixture method	Thresholding-based
Edge detection	Edge-based
Watershed	Edge-based
Region growing	Region-based
Fuzzy c-mean	Clustering-based
K-means	Clustering-based
Atlas-based segmentation	Other
Neural networks	Other

Table 3.1 *Popular image segmentation methods.*

3.3 Deep Learning in image analysis

Deep learning methods have become state-of-the-art methods in many domains such as speech recognition, visual object recognition, object detection or segmentation. [4] Especially in medical image analysis, deep learning is getting more and more popular. Currently deep learning methods, in particular convolutional neural networks, are used in many application areas such as neuro, retinal, pulmonary, digital pathology, breast, cardiac, abdominal and musculoskeletal. One reason for popularity of the deep learning methods is that in medical image analysis feature extraction plays a crucial role and deep learning lets the computer learn the features that optimally represent the data instead of humans. [53]

3.3.1 Basics of neural networks

The idea of the neural networks is to take a large number of training examples as an input and then develop a system that will predict the output based on the training examples. Basically, this system will learn from the training data in order to solve a specific problem. Neural networks are inspired by human central nervous system and an example of a very simple artificial neural network (ANN) is represented in Figure 3.4. [18] [4]

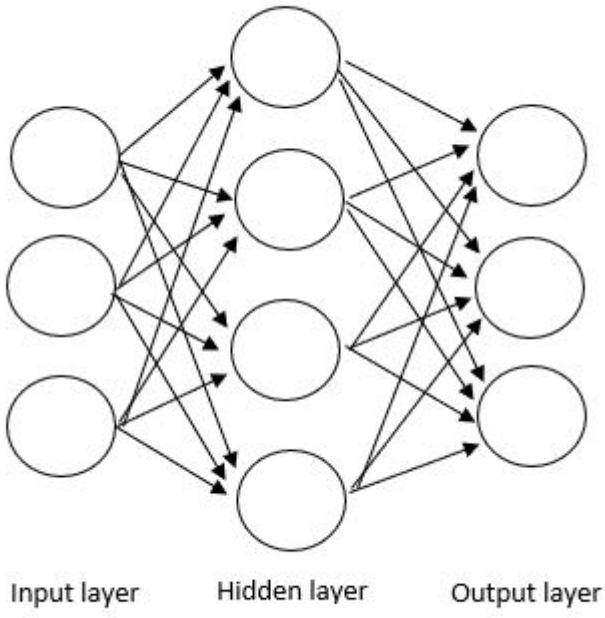

Figure 3.4 *Two layer ANN with one hidden layer and output layer. Circles are representing neurons in the network and arrows are connection between the neurons.*

From the Figure 3.4 we can see that the network consists of layers and neurons. Every layer in the network is actually a group of parallel non-connected neurons and the number of layers determines the depth of the network. Connections between the layers build the network. Inputs can be fed to the network through the first layer called input layer. Then input layer pushes values deeper to the network where layer of hidden neurons receives values as an input. Usually there are multiple hidden layers where outputs of the one layer are inputs of the subsequent layer. The last hidden layer finally feeds the final output layer. Each one of the layers can be specified in order to increase the learning accuracy. [14] [3]

ANNs are also called as feedforward neural networks and the simplest model is a multilayer percepton (MLP) network. Feedforward means that there are not any feedback connections in the system. When every neuron in one layer is connected to each neuron of the previous layer, network is called fully connected network. If network consists of at least two layers of neurons, it is called MLP network. When feedback connections are added to the network it is called recurrent neural network (RNN). [18] [4]

Neuron is also known as a perceptron and the basic structure is shown in Figure 3.5. Neuron takes one or more inputs and sums them to produce an output. Inputs

are usually multiplied with different weights and then passed through an activation function after summation. [4] One perceptron is represented mathematically as

$$y = f(\sum_{i=1}^{n} w_i x_i - b), \tag{3.7}$$

where x_i are input values, w_i corresponding weights and y output of the preceptron. Output is produced based on the transfer function f which is also called as activation function. Activation function can be linear or nonlinear and it maps the resulting values between the specified value range depending on the function of choice. Value b is in this case bias, which shifts the activation function $f(x)$ to the left or right. [4]

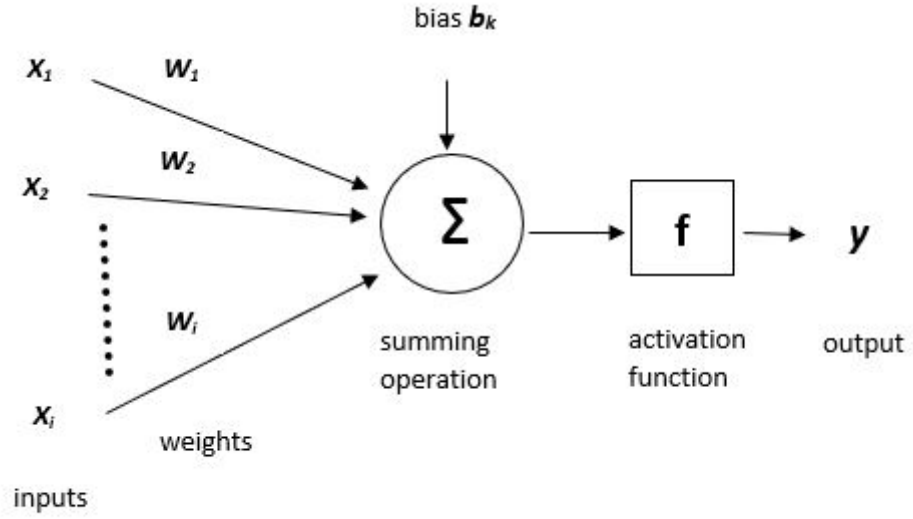

Figure 3.5 *Visualization of the perceptron.*

Often neural networks are working with very complicated and often nonlinear kinds of data such as images, audio or speech. We want our network to learn that data to be able to generate nonlinear mappings from inputs to outputs. That is the reason why mostly nonlinear activation functions are used. The most common activation functions are sigmoid or logistic functions, hyperbolic tangent and rectified linear units. [18] [4]

Sigmoid activation function is mostly used when output probabilities are predicted because it gets values between the range 0 to 1. Sigmoid is differentiable function which means that the slope of the curve can be found at any two points. Logistic sigmoid function is generally defined as

$$f(x) = \frac{1}{1 + e^{-x}}. \tag{3.8}$$

However, the fact that output is not zero centered is sometimes a problem. Sigmoid makes the gradient updates to go too far in different directions making it difficult to optimize. To overcome this problem hyperbolic tangent function may be used. Hyperbolic tangent is scaled and biased version of logistic sigmoid function and is defined with the function

$$f(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}}. \tag{3.9}$$

Both sigmoid and hyperbolic tangent suffer from vanishing gradient problem which means that gradients tend to get smaller and smaller during the training causing the front layers to learn very slowly. We focus more on training in section 3.3.2. Rectifier solves this problem and currently it is the most common and recommended activation function for deep neural networks. Rectifier is defined as

$$f(x) = \max(0, x). \tag{3.10}$$

Basically, rectifier thresholds activity at zero and units with rectifier functions are called rectified linear units (ReLUs). ReLUs are popular because they make deep convolutional networks train several times faster than networks with i.e. tanh units and faster learning influences the performance of large models. ReLUs are used only in hidden layers and for the output layer usually softmax function is used. Softmax function is defined as

$$f(x) = \frac{e^{x_j}}{\sum_{i=1}^{n} e^{x_i}}, \quad j = 1, ..., n. \tag{3.11}$$

Softmax function is a generalization of the logistic function and it computes the class probabilities for given input. Also, the softmax function can be used in various multiclass classification methods. [18] [4] [14]

3.3.2 Training the neural network

During the training, model parameters of a NN are optimized so that the error between the network outputs and prediction is as small as possible. This optimization

problem is solved by defining a loss function $J(\theta)$ and minimizing it. Loss function describes how correct the current model and weights are and can be typically written as follows

$$J(\Theta) = L(f(x; \Theta), y), \tag{3.12}$$

where L is per-example loss function, $f(x; \theta)$ predicted output, Θ model parameters, x input and y target output. In order to improve the model, the loss function is minimized using its gradients with respect to model parameters. Minimizing can be done using iterative process of gradient descent algorithm which updates the model weights in the opposite direction of the gradient of the loss function in order to find the global minimum. In deep learning applications, the most popular algorithm used in training is stochastic gradient descent (SGD) which is an extension to the gradient descent algorithm [4]. SGD performs a parameter update for every training example and is defined as

$$\Theta_{n+1} = \Theta_n - \eta \nabla J(\Theta; x(n), y(n)), \tag{3.13}$$

where Θ is model parameters, $x(n)$ and $y(n)$ are training examples, $\nabla J(\Theta; x(n), y(n))$ is gradient of the loss function and η is learning rate. Generally, each parameter update is computed with respect to a few training examples or mini-batch which decreases the variance and can lead to better convergence. [4] [14]

Learning with SGD can be sometimes a bit slow and to accelerate the learning process a momentum is introduced [4]. It gathers exponentially decaying moving average of past gradients and continues to move in the same direction. Momentum is defined as

$$v_t = \gamma v_{t-1} + \eta \nabla_0 J(\Theta) \tag{3.14}$$

$$\Theta = \Theta - v_t, \tag{3.15}$$

where γ is a momentum term and v_t is exponentially decaying avarage of the negative gradient which defines the direction and distance in which the model parameters move. [4]

In deep learning, model parameters are often updated during the training algorithm called backpropagation. This algorithm consists of two steps. First step is forward

propagation in which the input sum and output activation is calculated for each neuron and then stored for later use. Output activation is an output of the activation function when input sum of the neuron is feed to the function. Then backpropagation step propagates these output activations back to the network in order to compute the error for each neuron. What backpropagation actually does is that it computes the partial derivatives for single training example in order to find out how changing the model parameters change the loss function. First error of the final layer is calculated by computing the gradient of loss function with respect to outputs of the network. For the last layer in the network the error is calculated as

$$e_j^L = \frac{\delta l}{\delta a_j^l}\sigma'(z_j^l), \tag{3.16}$$

where L is loss function, a_j^l is the output of the neuron j in the layer l, σ is the activation function and z_j^l is the input sum of neuron j in layer l. For the other layers error values are defined as

$$e_l = ((\Theta^{l+1})^T e^{l+1}) \circ \sigma'(z^l), \tag{3.17}$$

where w is the weight matrix for layer l, e is error of the next layer, σ' is activation function and z is the input sum of layer l. The symbol $\circ$ represents Hadamard product in which two matrices are multiplied by elements. After the error of the last layer is computed it is possible to propagate backwards and calculate all the errors for every neuron in each layer. Then derivative of the loss function with respect to all weights is computed in the following way:

$$\frac{\delta L}{\delta \Theta} = a_{in} e_{out}, \tag{3.18}$$

where L is the loss function, a_{in} is the activations of the neuron inputs to the model weights Θ and error e. Then very similarly to the SGD the model weights are updated acording to the delta rule as follows:

$$\Theta_{n+1} = \Theta_n - \eta a_{in} e_{out}, \tag{3.19}$$

where Θ are model weights, η is learning rate, e is error and a_{in} is the activations of the neuron inputs. The backpropagate algorithm is a smart way to keep track on

small changes in weights as they propagate through the neural network and affect the loss. [4]

3.3.3 Regularization

Modern neural networks have a large number of weights which can cause a problem called overfitting. Overfitting means that the trained model describes the training data too accurately. Model learns the details and noise in the training data so well that it has a negative impact to the performance of the model with unseen data. Basically, model doesn't generalize enough from training data to unseen data. Overfitting is one of the main problems when training deep neural networks but luckily there are several ways to decrease that problem. One way to reduce overfitting is to increase the size of training data or reduce the size of the network, but there are also other regularization methods available. [4]

The most common regularization technique used to prevent overfitting is called $L2$ regularization or weight decay. This method moves the weights closer to zero by adding an extra term called regularization term to the loss function. This makes model to prefer small weights, but it has no impact on the optimum values since all the weights can be scaled down [58].

In deep learning one very effective regularization technique is called as dropout. The idea of the dropout is that at each training iteration dropout layer randomly removes neurons and its connections from the network. This makes neurons more robust and rely more on whole training data rather than one specific example. Studies have shown that dropout improves the performance of neural networks in most machine learning tasks making it very general technique. [55] On the other hand, dropout increases the training time of the model since the parameter updates are very noisy and the gradients that are being computed during the training are not gradients of the final architecture that will be used during testing. [4]

Training deep neural network is not an easy task. Each layer in the network gets its inputs from previous layer and this distribution changes during the training resulting slower training times by requiring lower learning rates. This problem is called as internal covariate shift and reducing it by normalizing layer inputs helps the training procedure. Batch normalization is one way to reduce internal covariate shift. It normalizes the input batch by subtracting the batch mean from the inputs and then dividing inputs by the batch standard deviation. Batch normalization has also a regularizing effect and in some cases it eliminates the need for dropout. [22] [33]

Data augmentation is also one way to make overfitting model perform better, especially when available data is very limited. For example for images, data augmentation techniques include rotation, scaling and translating image pixels. [60] It is also possible to interrupt the training when model's performance on validation set starts to drop during training. This method is called early stopping. In practice, the training is not actually stopped, instead, the model is saved at regular time intervals and finally the best candidate is picked. [4]

3.3.4 Convolutional neural network

Convolutional neural networks (CNNs) are special kind of neural networks even though they are very similar to ordinary neural networks. CNNs are made of neurons that have learnable weights and biases which are parallel in layers. Usually CNNs process data which comes in the form of multiple arrays such as time series data or image data. CNN architecture typically consists of convolutional layers, pooling layers and fully connected layers. Figure 3.6 represents a CNN which consists of two convolutional layers, two maxpooling layers, one fully connected layer and one softmax output layer.

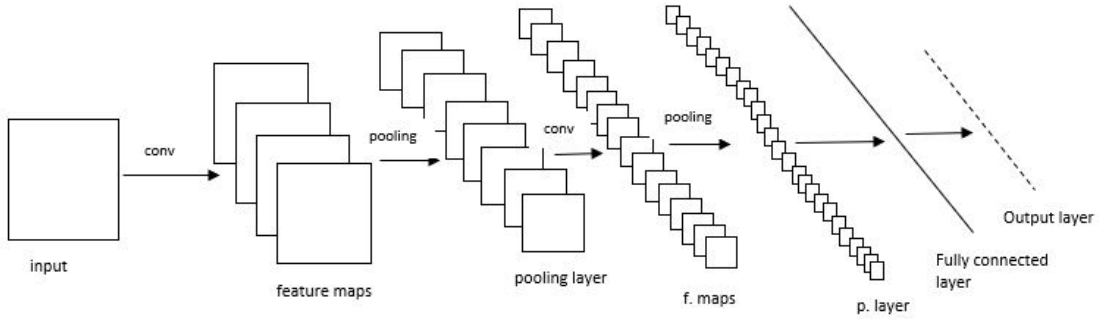

Figure 3.6 *Simple CNN consisting convolutional layers, maxpooling layers, fully connected layer and output layer.*

The term convolution in the network's name comes from the mathematical operation which CNN uses instead of general matrix multiplications. This convolutional layer is the core building block of a convolutional neural networks. Convolution for 1D signals is defined as

$$(f * g)(i) = \sum_{j=-\infty}^{\infty} g(j)f(i-j), \tag{3.20}$$

where f and g are input signals. The resulting output signal is a new signal which is produced by multiplying one signal by another delayed or shifted signal. In image processing, input signals are usually 2D images or 3D image volumes and convolution operation in neural networks is done in specific convolutional layer. In convolutional layer the first argument is an input, the second one is often called as a kernel and the resulting output signal is referred as a feature map because the purpose of the whole convolutional layer is to extract features from the input. 2D convolution is specified as

$$(f * g)(i, j) = \sum_m \sum_n f(m, n)g(i - m, j - n), \tag{3.21}$$

where f is 2-dimensional input image and g 2-dimensional kernel. Since CNNs are often used for segmentation tasks and in medical applications input batches are often 3-dimensional, 3D convolution is defined as

$$(f * g)(i, j, k) = \sum_m \sum_n \sum_l f(m, n, l)g(i - m, j - n, k - l). \tag{3.22}$$

The parameters of convolutional layer consist of learnable kernels and every kernel is spatially small but extends through the whole depth of the input. For example, the size of the typical kernel for the first convolutional layer is 5x5x3 in 3D case. For the images, kernel is slided over the width and height of the original image and for every position in the input image, the element wise multiplication is computed. As the kernel slides over the input image, a 2D activation map is produced and that map shows the responses of kernel in every position. During the learning process, the network will learn kernels that activate when they detect any kind of visual features such as edges, patterns and colors. However, the convolution of another filter over the same image gives as a result different feature map and the more filters are in convolutional layer the more image features get extracted. This improves the performance of the network. The number of kernels in convolutional layer defines the number of feature maps and finally these maps are stacked together in order to produce an output. Nonlinearity is applied after every convolutional operation in convolutional layers and this is usually done with ReLU function. [4]

The size of the resulting feature maps is defined by three parameters: stride, zero-padding and depth. Stride defines the number of pixels by which the kernel is slided over the input batch. For example, if stride is set 1 kernel moves 1 pixel at a time but with stride value 2 kernel jumps 2 pixels at a time. Basically, larger stride

values produce smaller feature maps. In zero-padding input is padded with zeros so that kernel can be applied to the bordering elements. With zero-padding size of the feature maps can be controlled. [28] [4]

Convolutional layers are often followed by a pooling layer which modifies the output of the previous layer. Idea of the pooling is to achieve spatial invariance by reducing the size of each feature map but keeping all the important information. Each pooled feature map corresponds to one feature map of the previous layer. This reduces the number of parameters in the network in addition to the reduced computation time. There are several functions to perform a pooling operation but the most common one is maxpooling. In maxpooling a filter is applied to the input batch by defining the spatial neighborhood and picking the maximum of the neighborhood within that window. The result is a new feature map which has lower resolution than the input. An example how to perform a maxpooling using 2x2 filter size is presented in Figure 3.7. [51]

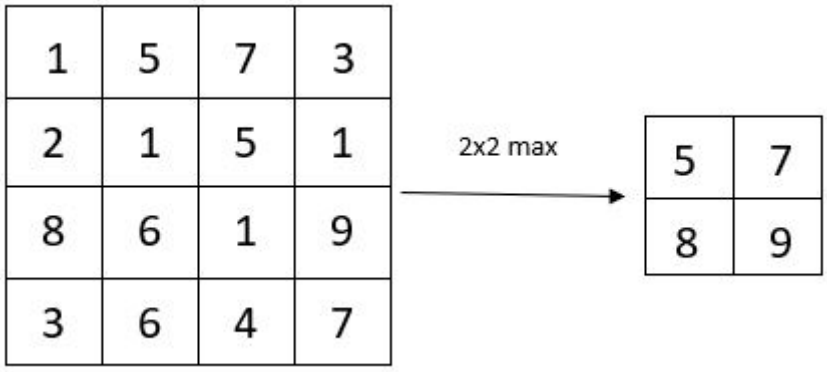

Figure 3.7 *Maxpooling performed using 2x2 max filter.*

The other suitable pooling functions are average pooling and L2-norm pooling. In average pooling average of the neighborhood is computed instead of picking the maximum element but these two other methods are rarely used. Using Convolutional neural networks in image classification and segmentation tasks has many benefits. In image analysis, the input batches can be very large containing even millions of pixels. Traditional networks apply matrix multiplications involving every input and parameter which leads to billions of computations. CNN focuses on the most important features in the image instead of specific pixel values and reduces the amount of computations by a large margin. Also, a CNN property called parameter sharing reduces the amount of memory needed in training. Parameter sharing simply means that weights are shared by all neurons for a specific feature map. [4] [3]

4. MATERIALS AND METHODS

In this chapter we go through the study work flow including data sets, data preprocessing, CNN architecture, training the CNN and testing the CNN. The whole work flow is visualized in Figure 4.1.

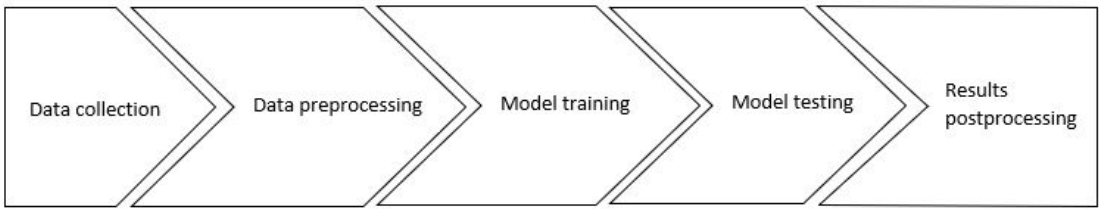

Figure 4.1 *Process work flow.*

First of all, data needs to be collected. In this case we are using one big data set containing images produced by multiple different MR sequences. Before training the CNN, all of the data needs to be preprocessed because images may have different resolutions, orientations or number slices in image volumes. Data is divided into subsets for training and testing the CNN. Preprocessed training data is feed to the predefined CNN and after decided number of epochs training is complete. Then test image data set is feed to CNN in order to get automatically produced segmentations. Finally result images are post-processed and numerical evaluation of the result images is performed.

4.1 Data

The available data consists of set of images from 606 patients obtained from the Leukoaraiosis and Disability (LADIS) study. LADIS study involved 11 European centers and its aim was to evaluate age-related white matter changes as a predictor of the transition to disability or death in elderly subjects that were followed up for 3 years. Studied patients aged between 65 and 84 years and they were investigated for complaints such as cerebrovascular events, problems with memory or motor, mood

alterations or other neurological problems. MRI scanning was performed according to the standard protocol in which 0.5-T or 1.5-T scanners were used. [43] All imaging parameters can be found from the Table 4.1.

Parameters	FLAIR	T1	T2
Echo time (TE)	100-140 ms	4-7 ms	100-120 ms
Repetition time (TR)	6000-10000 ms	10-25 ms	4000-6000 m
Inversion time	2000-2400 ms	-	-
Flip angle	-	15°-30°	-
Voxel size	1x1x1-1.5 mm^3	1x1x1-1.5 mm^3	1x1x5-7.5 mm^3
Number of slices	19-24	256	19-24

Table 4.1 *Imaging parameters used in LADIS study.*

A subset of 540 cases from 606 patients were used in training and testing the neural network. The missing 66 subjects didn't have T2-weighted images or failed preprocessing, so they were left out. For all of the 540 cases WMH and infarct lesions were delineated as different annotation classes by human expert using semiautomatic segmentation method. WMH were marked and borders were set using local thresholding on each slice and no difference between subcortical and periventricular hyperintensities was made [59]. Infarct lesions included both cortical and lacunar infarcts and they were delineated as different classes manually after WMH segmentation.

Tissue segmentation images containing white matter, gray matter and cerebrospinal fluid were obtained from T1 images using multi-atlas segmentation method proposed by Koikkalainen et al. [31]. In this method different brain structures were segmented from T1 images by registering T1 image and multiple atlases using non-rigid deformation. Then 12 atlases out of 60 atlases were selected in order to build a probabilistic atlas which was used as a prior in intensity based classification using EM algorithm [39]. Also, brain mask images were obtained from T1-weighted images. Brain mask images are binary images in which skull and other non-brain tissue such as eyes and dura in addition to background are labeled as 0. With the help of brain mask images, all non-brain tissue was extracted from images during image preprocessing.

White matter probability maps were acquired from 540 expert annotated WMH image segmentations. Every WMH segmentation image was registered to the T1 template which is an average image over several hundred cases including both healthy patients and patients with brain disorders. Then these template registered WMH

segmentations were divided into two subsets, both containing 270 images. These new subsets correspond to the training and testing sets in WMH segmentation task (see Table 4.2). Probability map for training set was made by summing all template registered segmentations from testing set together and dividing this image by the number of segmentations. Probability map for testing set was produced similarly but instead of summing testing segmentations together, training segmentations were summed together. The last step was to register probability map obtained from training set to the each FLAIR image in the testing set. Similarly, probability map obtained from testing set was registered to the each FLAIR image in the training set. Reason for this kind of probability map construction is to make sure that FLAIR registered probability map does not contain WMH segmentation obtained from the same FLAIR image. Images used for training and testing the models are visualized in Figure 4.2.

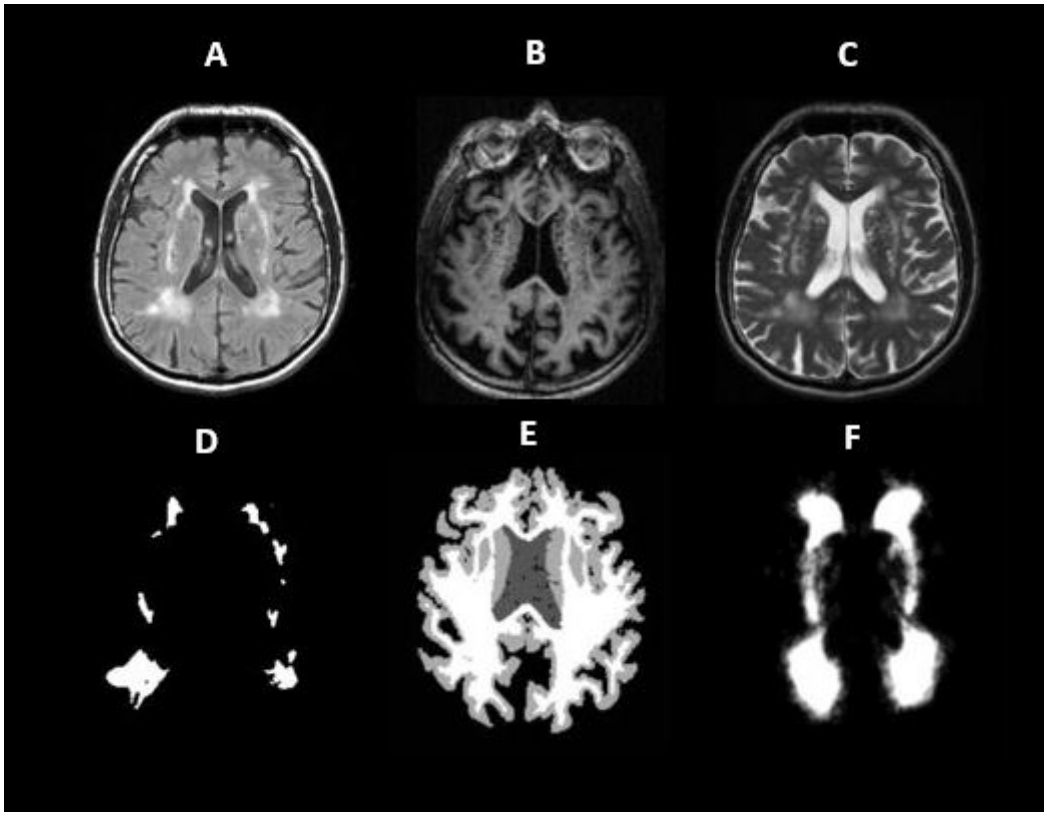

Figure 4.2 *Images used for training and testing. On the top row are FLAIR (A), T1-weighted image (B) and T2-weighted image (C). On the bottom row are WMH segmentation image (D), tissue segmentation image (E) and WMH probability map (F).*

4.1.1 Data sets

Out of 540 subjects a subset of 55 subjects contained cortical infarct segmentations and 210 lacunar infarct segmentations. Data sets for training and testing was organized so that for WMH segmentation task training set consisted 270 images and testing set 270 images. For WMH and cortical infarct segmentation task, modified 10-fold cross-validation was used so that every training set consisted 150 images including all images with cortical infarcts outside the testing set. In other words, 540 subjects were divided into 10 test sets. Then every test set had 54 images including 5 or 6 images with cortical infarcts. Training sets, on the other hand, had 150 images containing 50 or 49 images with cortical infarcts depending on the number of images with cortical infarcts in test sets.

For WMH and lacunar infarct segmentation task, modified 10-fold cross-validation was also used. In this case, training sets had 300 images containing 189 images with lacunar infarcts and testing sets 54 images including 21 images with lacunar infarcts. The final task was to segment all WMH, cortical infarcts and lacunar infarcts at the same time using modified 10-fold cross-validation. Training sets for that purpose were set at 300 images containing 49 or 50 images with cortical infarcts and 189 images with lacunar infarcts. Test sets contained 54 images including 5 or 6 images with cortical infarcts and 21 images with lacunar infarcts. After training and testing, all 10 test sets were combined in order to construct final result set containing 540 images. Reason for this kind of arrangement was lack of images with infarcts. Data sets can be found from the Table 4.2.

	WMH	WMH+CORT	WMH+LAC	WMH+CORT+LAC
Training set	270	150	300	300
Cortical infarcts (Training)	-	49 -50	-	49 - 50
Lacunar infarcts (Training)	-	-	189	189
Cortical infarcts (Testing)	-	55	-	55
Lacunar infarcts (Testing)	-	-	210	210
Testing	270	540	540	540

Table 4.2 *Data sets and number of images for every segmentation task. LAC means lacunar infarct and CORT cortical infarct.*

4.2 Preprocessing

Data must be preprocessed in order to get an accurate prediction using neural networks. Usually neural networks don't require much data preprocessing since the learning algorithms are able to work with the original data. However, in this case

there are six different image volumes obtained from the same subject and the orientation, number of slices and resolution varies depending on the used MR imaging sequence. The aim of the image preprocessing is to register all the image volumes together so that all volumes are overlaying in the same scene and the anatomical structures are located in the same location in every image. The image preprocessing pipeline is visualized in Figure 4.3.

MR imaging could be done in several different angles. Therefore, first step is to swap images to the desired orientation of the head so that nose is pointing in the same direction in every image. Since the images are acquired by using different MRI imaging sequences they also have different voxel sizes, resolutions and number of slices. In order to achieve best possible registration result all images are re-sliced into isotropic volumes so that axial slices have 1 mm in three dimensions. This is done by using linear interpolation which can be defined in 2D space as

$$\frac{y - y_0}{x - x_0} = \frac{y_1 - y_0}{x_1 - x_1}, \tag{4.1}$$

where (x_0, y_0) and (x_1, y_1) are known points.

Training and testing images for the 2D neural network are interpolated so that the resolution is 256x256 and the number of slices stays changeless. However, for the 3D neural network training and testing images, interpolation is done to the whole volume. The resolution is interpolated to 256x256 and the number of slices varies depending on the result of the isotropic shape-based interpolation.

WMH segmentation images are binary images in which regions containing WMH are labeled as 1 and background as 0. Before interpolation every segmentation image is multiplied by 100 and after the interpolation segmentation images are thresholded in order to get binary segmentation images which correspond to the isotropic FLAIR images. This is performed also for the segmentation images which contain multiple labels including cortical and lacunar infarcts. Multiplication and thresholding is done for every label separately. Brain mask images are obtained from T1-weighted images before image preprocessing and interpolation into isotropic volumes can cause some holes in the brain area. Those holes need to be filled before the registration process.

Registration is the most critical step of the preprocessing in which all images are transformed into the same coordinate system using affine registration which have 9 degrees of freedom (DOF) in three dimensions. [25] If registration is not performed properly, it will decrease the neural network's learning ability since the anatomical

structures are not located in the same spot in every image. As mentioned in the chapter 3.1, registration is an optimization problem. In this case, the image registration algorithm uses NMI for maximizing the similarity between the registered images.

Next step after all images are in the same coordinate system is to remove all unnecessary non-brain tissue which can decrease the neural network's ability to build the best possible model for detection of WMH and infarcts. Especially, eyes can have very similar intensity values compared to the regions with cortical infarcts and this will affect the neural network's ability to learn the features describing cortical infarcts. Tissue removal is performed using brain mask obtained from T1-weighted images and resulting skull stripped image contain only voxels which are labeled as non-background in brain mask image.

MRI images have often some corruption due to low frequency signal which is caused by inhomogeneities in the magnetic fields of the MRI machine. This corruption blurs images and reduces high frequencies along the image resulting intensity value changes in the same tissue. This so called bias field decreases the performance of the segmentation and classification algorithms, and it needs to be corrected. [26] Correction is done using N4ITK bias correction algorithm which is an improved version of the nonparametric nonuniform normalization (N3) algorithm [56]. It combines fast and robust B-spline approximation algorithm with a hierarchical optimization strategy and allows multiple resolutions to be used in during the bias field correction in order to achieve high-quality performance. Bias field correction is performed only for FLAIR, T1 and T2 images.

After the N4 correction FLAIR, T1 and T2 images are intensity normalized within same scale by intensity z-scoring which is defined as:

$$z = \frac{x - \mu}{\sigma}, \tag{4.2}$$

where μ is mean value and σ standard deviation of the area containing only brain tissue in the image. Finally, images are aligned into same orientation so that center point of each image is located in the same spot. This is done by estimating the center point of the brain area in the image and moving it to the center of the result image.

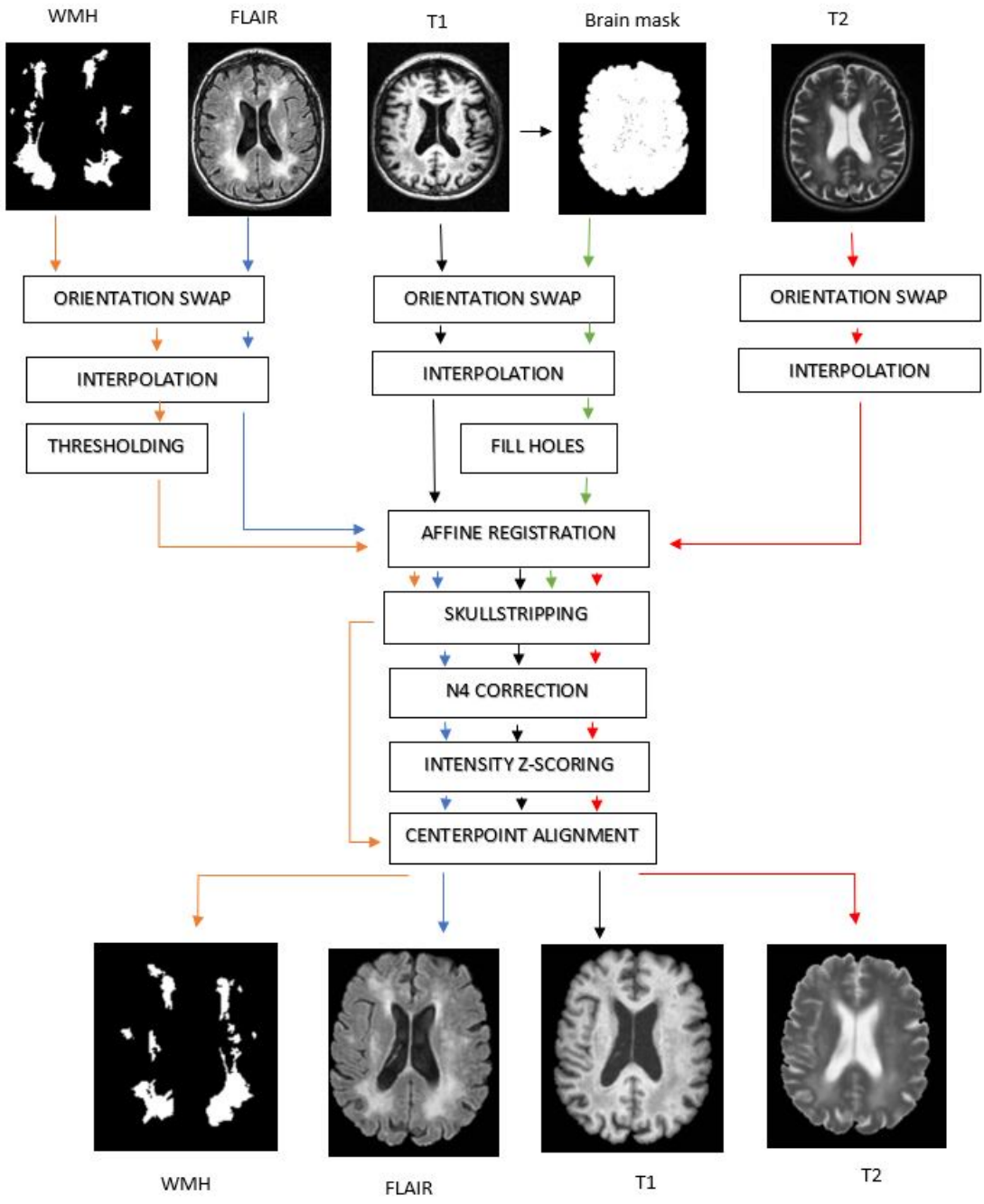

Figure 4.3 *Preprocessing pipeline visualized for FLAIR, T1, T2 and WMH segmentation images.*

4.3 CNN architecture

Convolutional neural network is multi-functional machine learning model that can be trained to recognize, segment and differentiate WHM-related lesions and infarcts from images and volumes. Typically, CNNs are used in classification tasks where

output of the network is single class label for the whole image. However, in this specific segmentation task the desired output should include localization and a class label is assigned to each pixel. Defining the CNN structure is important in order to achieve good segmentation results and aspects that need to take into account when choosing the network structure are for example network's field of view, receptive field and network's complexity.[56]

The chosen network used in this project is U-shaped network proposed by Guerrero et al. [16] which is previously used to segment WMH and infarct lesions. Even though fully connected layers are not used in this network architecture, the model is fully convolutional network. It is trained with large image patches which have high resolution allowing the network extract high and low-level features from image patches. The use of residual architecture improves the network's optimization convergence and it allows networks to be trained deeper. This architecture is designed for both 2D and 3D images and it is capable of processing multi-channel inputs such as T1, T2 and tissue segmentations in addition to FLAIR images. This so called uResNet for 2D image patches is visualized in Figure 4.4.

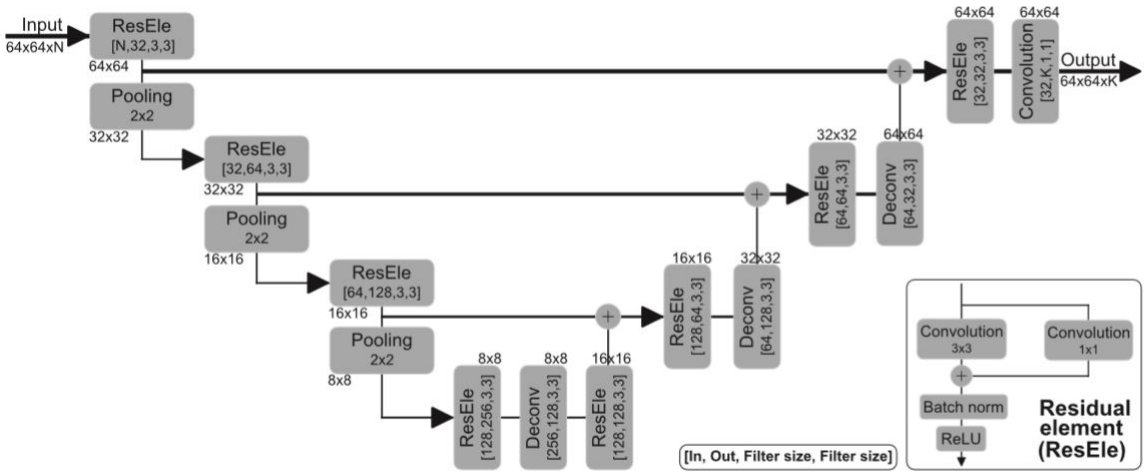

Figure 4.4 *uResNet architecture [16].*

The architecture consists of analysis path in which the left side of the path is contractive and is close to the typical architecture of the convolutional network. It is composed of 8 residual elements, 3 deconvolution layers and one convolutional layer that maps feature vectors to the desired number of classes. The left side of the network consists of repeated set of residual elements and each residual element is followed by 2x2 max pooling operation. Each step doubles the number of feature channels. Residual element consists of 3x3 convolution, 1x1 convolution, batch normalization and residual linear unit. Convolution results are added together, and the result is followed by a batch normalization and rectified linear unit as a non-linear

activation function. Residual elements allow signal to propagate from one block to other directly and this will improve network's generalization and makes training easier [19].

The right side of the uResNet is expansive and every step on this side of the network consist of repeated set of residual elements followed by deconvolution layer which halves the number of feature channels. Concatenation with residual elements and the corresponding feature map obtained from the prearranged layer from the left side of the network is used to skip connections with summations in order to reduce the complexicity of the network [16] and improve network's ability to localize [46]. The last layer which is 1x1 convolution operation maps the feature vector to the preferred number of classes and result is passed to the element-wise softmax function which calculates the class probabilities for every pixel or voxel.

4.4 Network training and testing

Neural networks were implemented in Python using Theano package and Lasagne which is a Theano based library for building and training neural networks [40]. Theano is a numerical computing library designed for Python and it allows processing with Nvidia graphics processing units by using computing toolkit CUDA (Compute Unified Device Architecture) and cuDNN (CUDA Deep Neural Network). CUDA is a computing platform created by Nvidia and it allows access to GPU's virtual instruction set and parallel computational elements. CuDNN is a GPU-accelerated library for deep neural networks providing faster neural network training [41]. Both Lasagne and CUDA support building two and three dimensional neural networks and in this paper both networks are built and trained.

In this work the way how training data is sampled and used as an input to the neural network needs to be considered carefully. Because the number of the healthy tissue or background voxels in the training images is larger than the tissue labeled as WMH or infarcts, the class imbalance needs to be taken into account. This class imbalance problem is solved by applying the patch sampling. In patch sampling, samples are extracted only from the locations centered by WMH or infarct tissue. This, however, will result in location bias where WMH or infarct voxels are expected to be in the center of each sample due to similar size of the sample and the field of view of the neural network. This bias is solved by applying a random shift Δx,Δy to random subset of WMH and infarct voxels in order to augment the data set. [16] This is visualized in Figure 4.5. Training patches of 64x64 for 2D training and 64x64x32 for 3D training were extracted from those augmented samples. For WMH segmentation task, image patches and their corresponding labels were extracted from random

volumes so that subset of all possible locations were labeled as WMH. However, when segmenting both WMH and cortical or lacunar infarcts random subsets of 20 % for WMH and 80 % for infarcts is used to prevent class imbalance problem. Also, when segmenting WMH, lacunar infarcts and cortical infarcts, subsets of 10 % for WMH, 30 % for lacunar infarcts and 60 % for cortical infarcts were used.

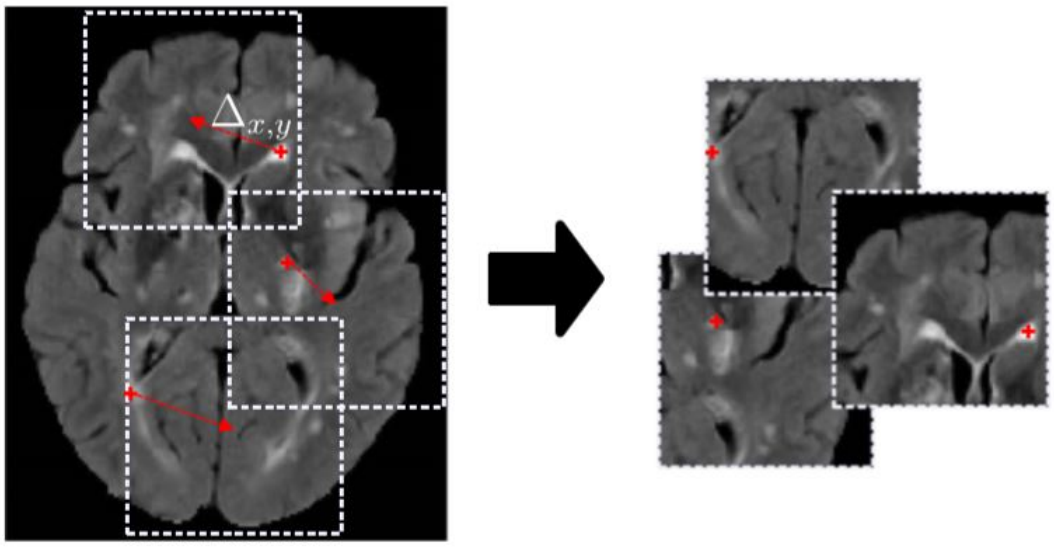

Figure 4.5 *Patch sampling. Random shift applied to random subset of WMH voxels. [16]*

Since uResNet is trained by backpropagation algorithm one of the most important decisions when training neural network is to choose correct loss function. Guerrero et al. [16] compared performance of the four different loss functions including classical cross-entropy, bootstrapped categorical cross-entropy, pseudo dice coefficient and weighted cross-entropy for WMH segmentation task. Experiment was done using similar uResNet CNN architecture than we are using in this study. FLAIR images were used as an input and Dice scores on the whole brain area were calculated for evaluating the CNN. Based on those study results classical cross-entropy was achieving the best performance and it was chosen as a loss function in this work. Classical categorical cross-entropy is defined as

$$H = -\sum_{n=1}^{N} y_n \log(f(\theta, x_n)), \tag{4.3}$$

where $f(\theta, n)$ is the model mapping function and N is the number of voxels. Vectors x_n and y_n are full of zeros except for at one position which represents the current

labels.

During the training network weights are updated using Adam optimization algorithm [48] which is slightly different from momentum introduced in section 3.3.2 . Adam optimizer algorithm is a gradient descent optimization algorithm which computes adaptive learning rates for each parameter. Adam also stores exponentially decaying average of past squared gradients v_t in addition to exponentially decaying average of the past gradient m_t similarly to momentum (see equation 3.11):

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1-\beta_2) g_t \\ v_t &= \beta_2 v_{t-1} + (1-\beta_2) g_t^2, \end{aligned} \tag{4.4}$$

where β_1 and β_2 are decay rates g past gradients, m_t estimate for first moment and v_t for second moment. These estimates are biased towards zero and they need to be corrected:

$$\begin{aligned} \hat{m}_t &= \frac{m_t}{1-\beta_1^t} \\ \hat{v}_t &= \frac{v_t}{1-\beta_2^t}. \end{aligned} \tag{4.5}$$

This leads to Adam parameter update rule which can be defined as

$$\Theta_{t+1} = \Theta_t - \frac{\eta}{\sqrt[2]{\hat{v}_t} + \epsilon} \hat{m}_t, \tag{4.6}$$

where Θ is network's weights and η learning rate.

Learning rate which defines the step size towards the global minimum is set at 0.0005. Higher and lower learning rates were also tried but their performance was worse compared to value 0.0005. Other learning parameters were set so that β_1 was 0.9, β_2 was 0.999 and ϵ was 10^{-8}. These parameters are the default values proposed by the authors of the Adam optimization algorithm [30].

Overfitting is a common problem for neural networks as discussed in chapter 3.3.2. In order to prevent overfitting problem, a dropout and batch normalization layers are used. Dropout layers are placed just before deconvolution layers on the right-hand side of the uResNet. Effect of the L2 regularization to networks learning ability was

also studied, but from the Figure 4.6 can be seen that L2 didn't improve networks performance and training was done without L2 regularization. Dice results visualized in the Figure 4.6 are calculated on the whole brain MR volumes.

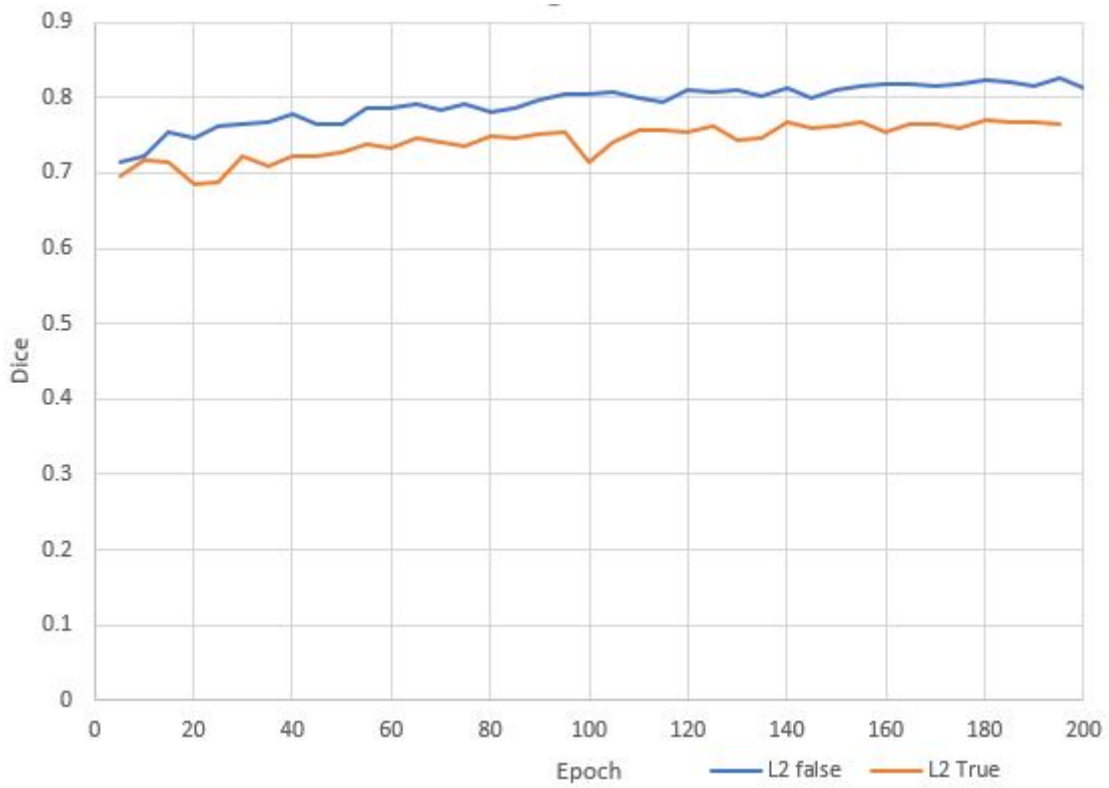

Figure 4.6 *Effect of L2 regularization. Dice scores during training without L2 regularization (blue) and Dice scores with L2 regularization (yellow).*

Training is done in epochs and every epoch consists of subepochs which are formed by multiple patches. Every patch is processed independently by the CNN and one batch consists of different random segments extracted from image volumes. In this work one epoch consists of only one subepoch which is formed by 100 patches and size of the one patch is 32 random segments. Number of epochs were set at 300 for WMH segmentation task and 500 for WMH and infarct segmentation task. After every fifth epoch, a test with testing set is performed in order to visualize the learning process using the Dice score between the expert annotated segmentation and neural network's segmentation for both training set and testing set.

Training takes a lot of time and the most important factors which affect CNN training times are number of epochs, number of input channels, number of patches and is model trained with 2D or 3D random segments. Examples of average training times for six different CNN models are presented in Table 4.3. Number of epochs in this case was 300.

Model	Input channels	Training time
WMH (2D)	FLAIR	3 h 33 min 21 s.
WMH (2D)	FLAIR-T1	3 h 54 min 52 s
WMH (2D)	FLAIR-T1-tissue	4 h 32 min 17 s
WMH (3D)	FLAIR	24 h 54 min 3 s
WMH (3D)	FLAIR-T1	34 h 3 min 36 s
WMH (3D)	FLAIR-T1-tissue	42 h 46 min 2 s

Table 4.3 *Training times for 2D and 3D CNN models designed for WMH segmentation using different input channel sets (FLAIR, T1 and tissue segmentations).*

When training is complete and test images are feed to the trained model, the output segmentation is produced in seconds.

4.4.1 Post-processing and validations

The final evaluation of CNN's performance is done visually in addition to calculating the Dice scores and correlations between the segmented images and the expert annotated images for both WMH and infarct volumes. Also, sensitivity of the infarct detector, number of false positive segmented infarcts and differences between the automatically produced segmentations and expert annotated images are also determined. However, evaluating is not performed for the unprocessed result images, instead, result images are processed back to the original space. In other words, preprocessing step is performed backwards in order to compare result segmentations to the original expert annotated segmentations.

Firstly, center points of the brain area were shifted back to their original positions. Then result segmentation images were resized according to the original FLAIR image using linear interpolation. Registration was not needed in this case because result images were already in the same coordinate system with original FLAIR image. Only their size was different due to the isotropic interpolation step during preprocessing. Finally, result segmentation images were swapped to the desired orientation of the head based on original FLAIR image.

Dice similarity index or Dice score is a common way to measure the overlap between the result images and expert annotated images. It can be defined as

$$\frac{2 \times TP}{FP \times FN \times (2 \times TP)}, \tag{4.7}$$

where TP is number of true positives, FP number of false positives and FN number of false negatives. In the literature the Dice score of 0.7 or higher is considered as a good segmentation. [6]

Correlation estimates the statistical dependence between two populations. For example, Pearson correlation coefficient between expert annotated WMH volumes X and automatically produced WMH segmentations Y is defined as

$$R = \frac{\mathrm{cov}(X, Y)}{\sigma_x \sigma_y}, \tag{4.8}$$

where σ_x is standard deviation of X, σ_y is standard deviation of Y and $\mathrm{cov}(X, Y)$ covariance of X and Y. Correlation coefficient values approaching 1 have strong relationship, values approaching -1 have strong inverse relationship and values close to 0 have weak relationship. [17]

The sensitivity measures the ability to correctly detect infarcts from brain images. Sensitivity is defined as

$$Sensitivity = \frac{TP}{TP + FN}. \tag{4.9}$$

If sensitivity achieves 100 %, it means that all infarcts were detected.

Differences between the automatically produced segmentations and expert annotated images were visualized by creating distribution and difference images for WMH and infarcts. For example, WMH distribution image was created from WMH segmentations by registering all segmentation images to the same template. Then all segmentations were summed together in order to produce the distribution image.

5. RESULTS

Study results are divided into several different groups. White matter hyperintensities were segmented using both 3D and 2D convolutional neural networks with five different input channel sets. Infarcts in addition to the WMH were segmented using only 2D convolutional neural network. For infarct segmentation, only 2D network was used because during the preprocessing some of the small lacunar infarcts, which appeared only in one slice, were labeled as background when performing the isotropic interpolation into 3D volume. Three different WMH and infarct segmentation studies were performed as cortical and lacunar infarcts were segmented separately in addition to WMH. Finally, all cortical infarcts, lacunar infarcts and WMH were segmented at the same time.

5.1 White matter hyperintensity segmentation

Data consists of 540 images divided into subsets of 270 for training and 270 testing (see Table 4.2). For every test set, Dice scores for WMH and correlation (R) between the expert annotated WMH volumes and automatically produced WMH volumes were calculated. The model trained only with FLAIR images as an input achieved 0.672 Dice score and 0.982 correlation. When T1 images are added to the training, calculated Dice score increased to 0.696. Instead of training the model with FLAIR images, only T1 images were used as an input and this model achieved 0.321 Dice score and 0.917 correlation. All results for five different input channel sets obtained by using 3D neural network can be found from the Table 5.1.

WMH segmentation was also performed using 2D neural network. Table 5.2 shows results obtained by using 2D neural network when training and testing sets were identical to the sets used in 3D study. For example, the model trained with FLAIR and T1 images obtained 0.727 Dice score and 0.976 correlation, whereas the model trained only with T1 images achieved 0.514 Dice score and 0.908 correlation.

3D		
Channels	Dice (WMH)	R (WMH)
FLAIR	0.672	0.982
FLAIR-T1	0.696	0.982
FLAIR-T1-tissues	0.693	0.977
FLAIR-T1-Prob	0.696	0.982
T1	0.321	0.917

Table 5.1 *White matter hyperintensity dice scores and correlations WMH volumes using 3D convolutional neural network. Input channel sets include FLAIR images, T1-weighted images, tissue segmentation images (tissues) and WMH probability maps (Prob).*

2D		
Channels	Dice (WMH)	R (WMH)
FLAIR	0.693	0.969
FLAIR-T1	0.727	0.976
FLAIR-T1-tissues	0.676	0.97
FLAIR-T1-Prob	0.687	0.982
T1	0.514	0.908

Table 5.2 *White matter hyperintensity dice scores and correlations for WMH volumes using 2D convolutional neural network. Input channel sets include FLAIR images, T1-weighted images, tissue segmentation images (tissue) and WMH probability maps (Prob).*

In order to visualize WMH segmentation results, WHM distribution map of 270 patients was created from the result images. Result images were obtained by using 3D neural network with FLAIR and T1 images as inputs. Results were registered into same coordinate system, summed and the resulting distribution image is an average over the 270 image test set. White matter hyperintensity distribution is visualized in Figure 5.1.

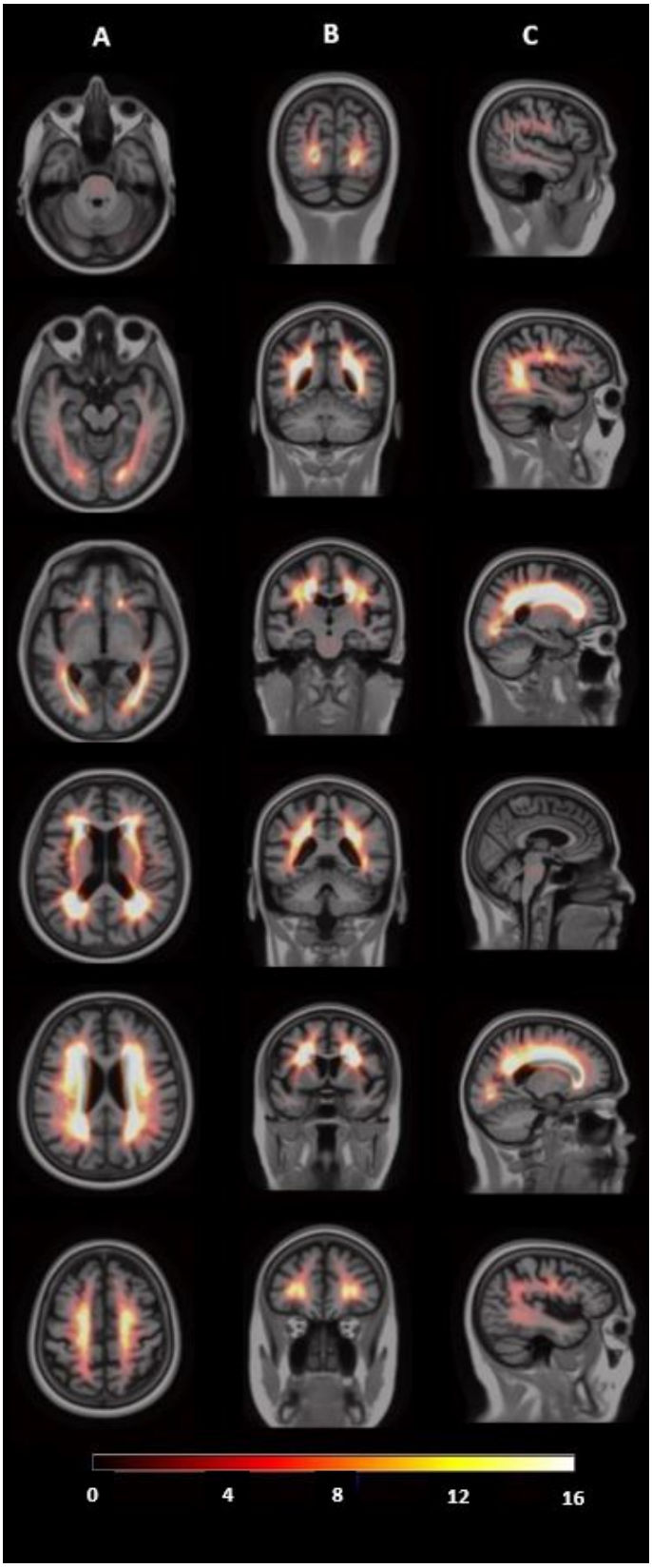

Figure 5.1 *WMH distribution of 270 test patients. Column A presents axial plane, column B coronal plane and column C sagittal plane.*

5.2 White matter hyperintensity and infarct segmentation

White matter hyperintensity and infarct segmentation was performed by using only 2D neural networks. Training was done using four different input channel sets including FLAIR, T1, T2 and tissue segmentation images. However, WMH and lacunar infarct segmentation study was performed only with three different input channel sets. Result images were evaluated by calculating dice scores and correlations for WMH and infarct volumes. In addition to the dice and correlation, the number of detected infarcts were calculated, and the performance of the candidate detector was measured by determining the detector sensitivity. This was done by comparing the amount of found infarcts from the result images to the number of infarcts in expert annotated images. Also, number of false positive segmentations per slice was calculated for every result image set.

Data set for training the network designed for segmenting WMH in addition to cortical infarcts consist of 150 images and was tested with 540 images (see Table 4.2). Dice scores, correlations, sensitivities and false positives for four different input channel sets can be found from the Table 5.3.

White matter hyperintensities and cortical infarcts				
	FLAIR	FLAIR-T1	FLAIR-T1-tissues	FLAIR-T1-T2
Dice (WMH)	0.756	0.757	0.758	0.765
Dice (Cortical)	0.22	0.290	0.317	0.351
Sensitivity (Cortical)	0.629	0.694	0.703	0.710
FP per slice (Cortical)	0.048	0.064	0.068	0.060
R (WMH)	0.982	0.985	0.986	0.987
R (Cortical)	0.619	0.319	0.328	0.682

Table 5.3 *Results for white matter hyperintensity and cortical infarct segmentation task.*

Data set for training the network for WMH and lacunar infarct segmentation consists of 300 images and the model was tested with 540 images (see Table 4.1). Test set was identical to the one used in WMH and cortical infarct segmentation study. Dice scores, correlations, sensitivities and false positives for three different input channels sets can be found from the Table 5.4. Instead of training the model with four different input channel sets (FLAIR, FLAIR-T1, FLAIR-T1-tissues, FLAIR-T1-T2), input channel set which used only FLAIR images was left out because some of the lacunar infarcts were not clearly visible in FLAIR images.

White matter hyperintensities and lacunar infarcts			
	FLAIR-T1	FLAIR-T1-tissues	FLAIR-T1-T2
Dice (WMH)	0.757	0.757	0.774
Dice (Lacunar)	0.287	0.289	0.294
Sensitivity (Lacunar)	0.503	0.503	0.479
FP per slice (Lacunar)	0.047	0.047	0.033
R (WMH)	0.978	0.978	0.970
R (Lacunar)	0.794	0.794	0.768

Table 5.4 *Results for white matter hyperintensity and lacunar infarct segmentation task.*

Finally, a model designed for detecting both lacunar and cortical infarcts in addition to WMH was trained using 300 training images and tested with 540 images (see Table 4.2). Test set was identical to the one used in previous WMH and infarct segmentation studies. Dice scores and correlation of WMH and infarct volumes were computed in addition to the sensitivity of the infarct detector. As in previous study, the number of false positives was calculated from the result images. Results can be found from the Table 5.5.

White matter hyperintensities, cortical and lacunar infarcts				
	FLAIR	FLAIR-T1	FLAIR-T1-tissues	FLAIR-T1-T2
Dice (WMH)	0.717	0.725	0.719	0.751
Dice (Cortical)	0.162	0.174	0.238	0.241
Dice (Lacunar)	0.209	0.205	0.214	0.281
Sensitivity (Cortical)	0.484	0.468	0.548	0.565
Sensitivity (Lacunar)	0.209	0.205	0.214	0.522
FP per slice (Cortical)	0.022	0.029	0.031	0.021
FP per slice (Lacunar)	0.150	0.169	0.196	0.150
R (WMH)	0.981	0.979	0.973	0.978
R (Cortical)	0.676	0.550	0.37	0.545
R (Lacunar)	0.506	0.493	0.384	0.636

Table 5.5 *Results for white matter hyperintensity, cortical infarct and lacunar infarct segmentation task.*

Dice scores used in these studies are averages over the whole result image set and they don't tell how the amount of WMH labeled tissue in a sample image affect Dice scores. In order to study the distribution of the WMH Dice scores, Dice scores with

respect to the WMH volume sizes are visualized in Figure 5.2. Result images were produced by the model designed for WMH and lacunar infarct segmentation. It was trained using FLAIR, T1 and T2 images as inputs and resulting Dice score for the whole image set was 0.751 (see Table 5.4)

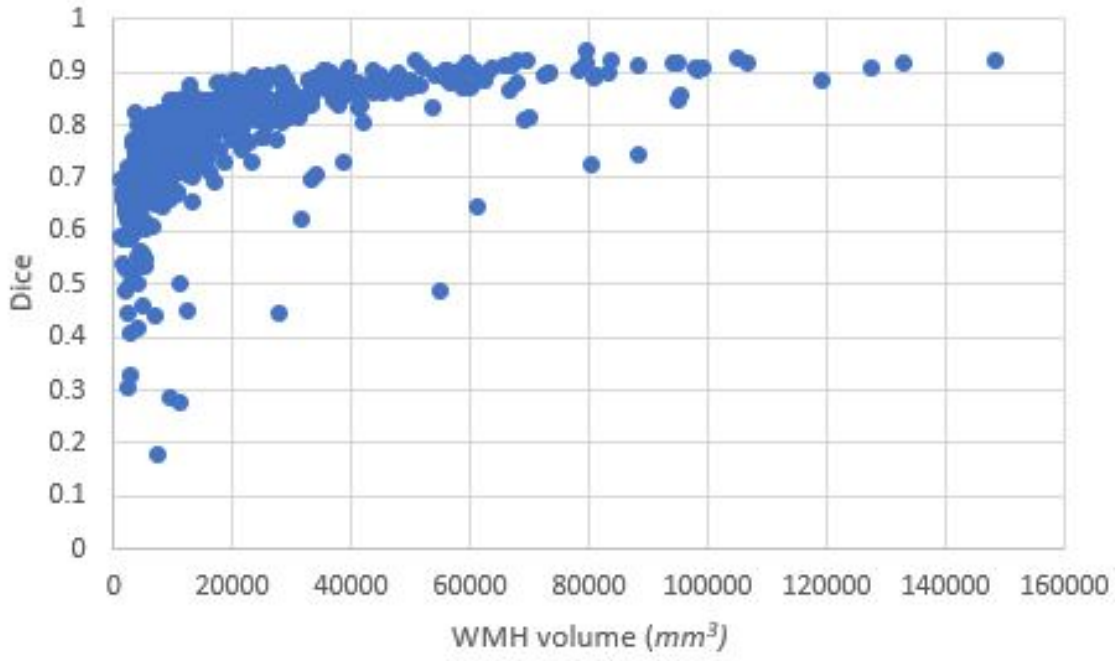

Figure 5.2 *WMH Dice scores and corresponding WMH volumes. The model was trained using FLAIR, T1 and T2 images as inputs.*

It is possible to subtract WMH labeled voxels of automatically generated images from the expert annotated images. This creates an image where locations of the false negative samples are visualized. When performed for the whole result image set, distribution image of the false negative WMH is created if subtraction images are registered into same coordinate system and summed together. This process can be performed also for the false positive WMH, false negative lacunar infarcts and false positive lacunar infarcts. Distribution images created from the result images obtained by using the model designed for WMH and lacunar infarct segmentation is shown in Figure 5.3. Input channels in this case were FLAIR, T1 and T2 images.

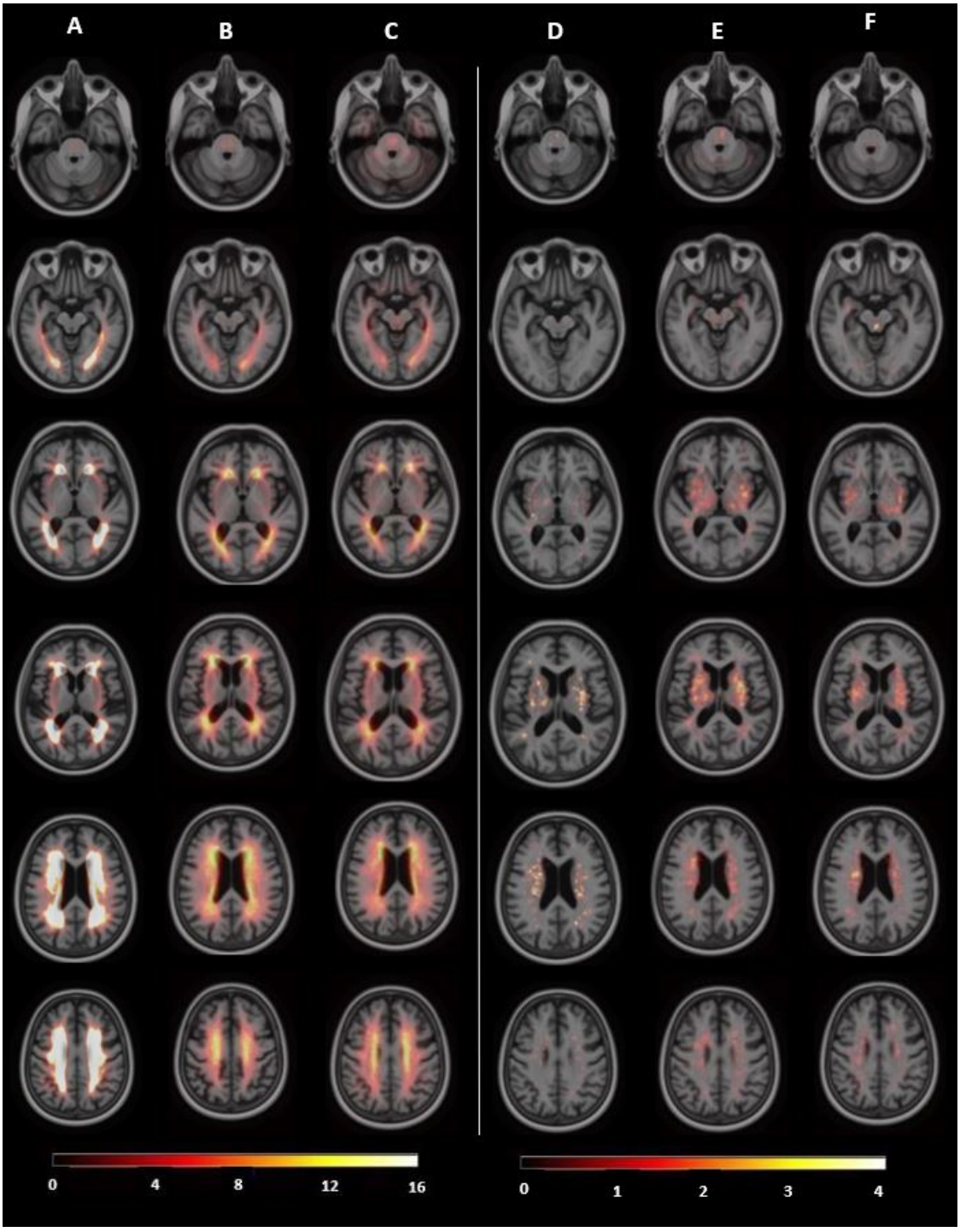

Figure 5.3 *Distribution images compared to the error images. A. WMH distribution, B. false negative WMH distribution, C. false positive WMH distribution, D. lacunar infarct distribution, E. distribution of the false negative lacunar infarcts and F. distribution of the false positive lacunar infarcts.*

6. DISCUSSION

The results show that uResNet is a powerful tool for segmenting WMH from the brain images because achieved WMH Dice scores are over 0.70 for most trained models. In the Figure 6.1 a randomly selected example of the segmentation result is visualized for WMH segmentation task obtained by using both 2D and 3D neural networks. Figure 6.1 also shows the expert annotated segmentation image in addition to the corresponding FLAIR image. It can be seen from the figure that automatically produced segmentations are very similar to the expert annotated segmentation and it is very hard to tell which solution is the best one.

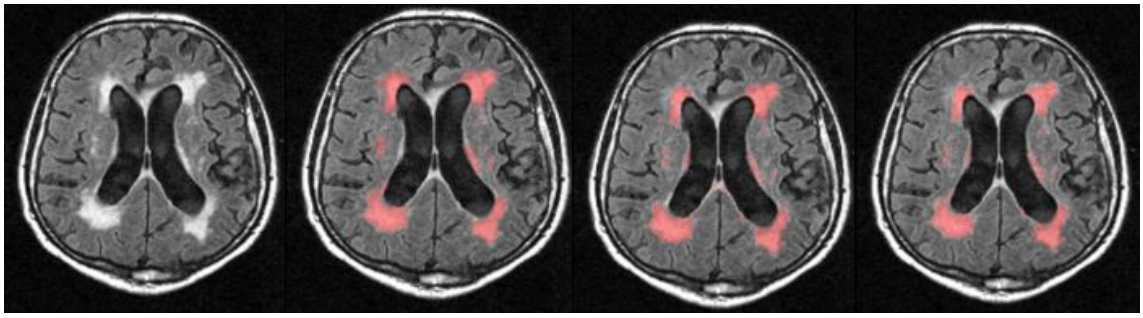

Figure 6.1 *WMH segmentations from left to right: FLAIR image, expert annotated segmentation, automatically produced segmentation using 3D CNN, automatically produced segmentation using 2D CNN.*

Tables 5.1 and 5.2 show the numerical evaluation of the WMH segmentation using either 2D or 3D uResNet. In the tables the Dice scores and correlations are listed depending on the different input channel sets used in training and testing the models. For 3D CNN, the best performing models achieved the average Dice score of 0.696 and 0.982 correlation over the whole test image set. There were two models achieving these scores and one was trained using FLAIR and T1 images as inputs while the other was trained using WMH probability maps in addition to the FLAIR and T1 images. When compared to the 2D models trained with identical data sets, the 3D results were not better. Instead, 2D model trained with FLAIR and T1 images achieved 0.727 Dice score outperforming all 3D models. This was a little surprise because adding the third dimension allows the surroundings on the axial plane influence the classification of each voxel and should improve the segmentation

results [27]. One possible reason for worse performance of the 3D model is that FLAIR images had thick slices (see Table 4.1) which can cause partial volume effect. Partial volume effect means that several different tissues are averaged together in a slice. Therefore, the information on z-axis is not accurate enough to increase the 3D model performance compared to 2D model.

Like mentioned in chapter 2, WMH are most visible in FLAIR images. However, in this study complementary information was added by training the models with multiple input channels. WMH segmentation was performed using input channels containing FLAIR, T1, tissue segmentation images or WMH probability maps. Even though the best performing model was trained with FLAIR and T1 images, testing data results (see Tables 5.1 and 5.2) show that difference between feeding only FLAIR images to the network compared to using multiple input channels is small. The exception is model trained with T1-images which achieved poor dice scores compared to the other models. This is understandable since WMH regions are not so clearly recognizable from T1 images compared to the FLAIR images. However, correlations were over 0.9 for all 2D and 3D models. Furthermore, adding more input channels require additional MR image acquisition and preprocessing. Also, additional input channels increase the time required to train CNN.

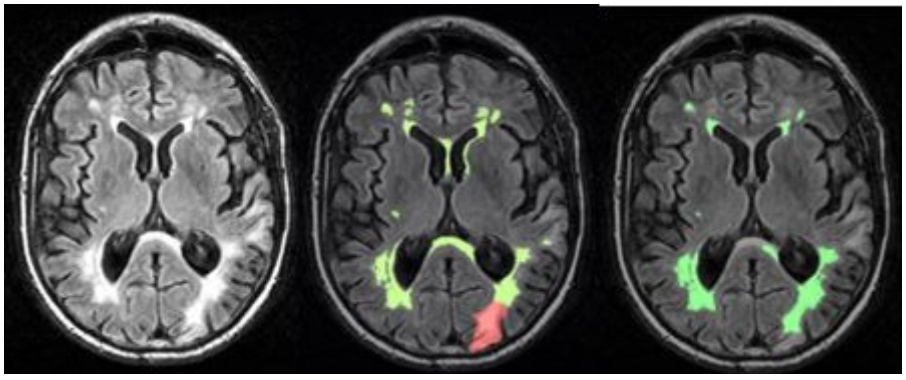

Figure 6.2 *WMH and cortical infarct segmentations. From left to right are FLAIR image, expert annotaded segmentation and automatically produced segmentation presented. Green regions are segmented WMH and red regions are segmented infarct lesions.*

The WMH segmentation results show that it is extremely difficult is to differentiate between WMH and infarct lesions. Especially, cortical infarcts which may appear in FLAIR images as large hyperintense regions are often labeled as WMH. This can be seen in Figure 6.2. In addition to the possible improvements in WMH segmentation, detection of the infarcts is clinically important. Therefore, cortical and lacunar infarcts were segmented alongside WMH. Training and testing sets were constructed from FLAIR, T1, T2 or tissue segmentation images. The numerical evaluating was

performed by computing WMH and infarct dice scores, correlations, detector sensitivities and amount of false positive samples per slice. Results for three different studies can be found from the Tables 5.3, 5.4 and 5.5.

For WMH and cortical infarct segmentation task (see Table 5.3) the highest Dice score 0.765 was obtained using the model which was trained with FLAIR, T1 and T2 images. Dice scores for other models were also relatively high: scores were over 0.75 for every channel input set. Also, WMH correlations were close to one in every case. Cortical infarct segmentations, on the other hand, left a lot room for improvement. The highest Dice score was 0.351 and sensitivity 0.71, both of which were obtained using model trained with FLAIR, T1 and T2 images. From the results we can see that adding additional input channels increases the segmentation accuracy for cortical infarcts and increases also the detector sensitivity. However, cortical infarcts are usually very noticeable and relatively large regions and if 3 out of 10 cortical infarcts were not detected, performance is poor. Especially, older chronic infarcts which appear as darker regions where brain tissue has died are usually left undetected. This is visualized in Figure 6.3 where one good segmentation and one bad segmentation are presented. The main reason for poor performance could be the lack of training data in which only 55 images contained cortical infarcts.

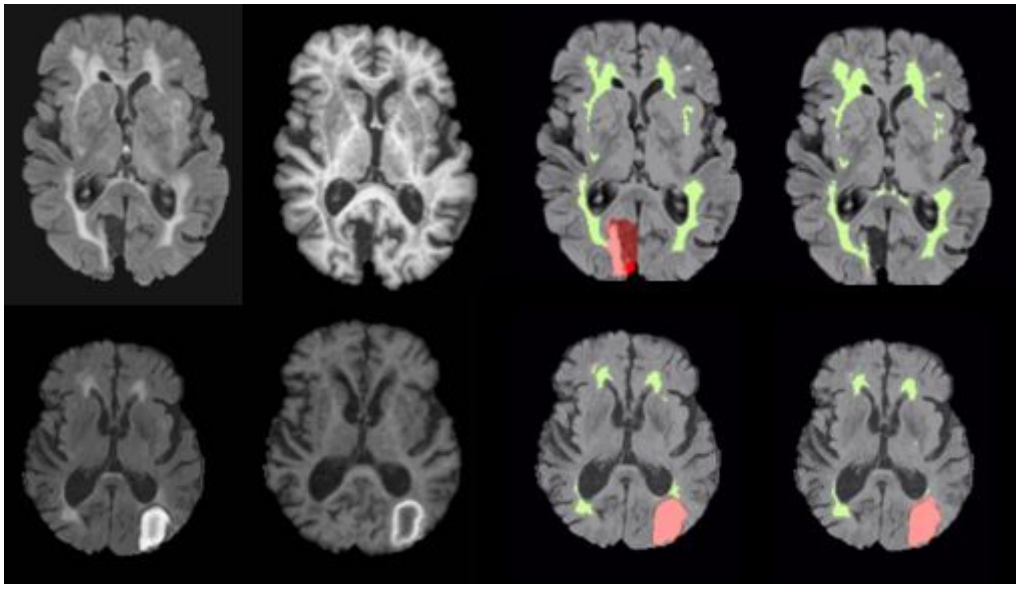

Figure 6.3 *Examples of WMH and cortical infarct segmentations. One of the worst segmentations on the top row and one satisfying segmentation on the bottom row. Images from left to right are FLAIR image, T1 image, expert annotated segmentation and automatically generated segmentation image. Red regions are infarct lesions and green regions are WMH.*

Lacunar infarcts were also segmented alongside with WMH and the results for numerical evaluation can be found from the Table 5.4. Adding lacunar infarcts into

training set seemed to increase WMH segmentation performance and the best performing model achieved 0.774 average Dice score and 0.97 correlation. This model was trained using T1 and T2 images in addition to the FLAIR images. Other models performed also very well achieving 0.757 Dice scores and 0.978 correlations which are even better compared to the model trained with FLAIR,T1 and T2 images. Models trained with either FLAIR and T1 images or FLAIR, T1 and tissue segmentations images achieved 0.503 sensitivity for detecting lacunar infarcts. Adding T2 images to the training data didn't increase the sensitivity of the detector, instead, sensitivity dropped from 0.503 to 0.479. However, adding T2 images reduced the amount of false positive segmented lacunar infarcts obtaining only 0.033 false positives per slice.

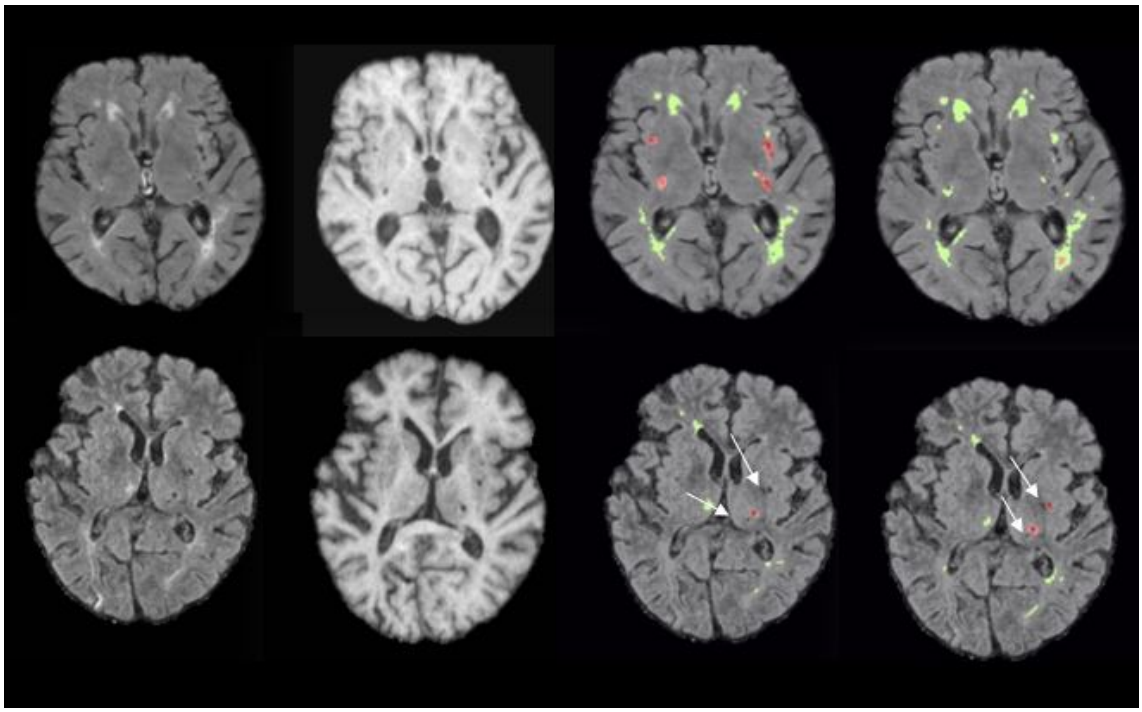

Figure 6.4 *Examples of WMH and lacunar infarct segmentations. One of the worst segmentations on the top row and on the bottom row is one example in which it is very difficult to say which segmentation is the best one. Images from left to right are FLAIR image,T1 image, expert annotated segmentation and automatically generated segmentation image. Red regions are infarct lesions and green regions are WMH.*

One of the biggest difficulties for the model was to differentiate between the lacunar infarcts and perivascular spaces which are very similar-looking structures. They are also filled by cerebrospinal fluid and can be up to 10 to 20 *mm* large. This can be seen in the Figure 5.3 where distribution of FN and FP lacunar infarcts are visualized. It is noticeable that most of the FP lacunar infarct segmentations focus on the thalamus, internal capsule, external capsules and basal ganglia, where perivascular spaces usually appear [12]. Also, the distribution of FN lacunar infarct

segmentations look very similar to the distribution of lacunar infarcts (see Figure 5.3). Some differences between the automatically produced segmentations and expert annotated segmentations are visualized in Figure 6.4.

Even though WMH segmentation study is not directly comparable with WMH and infarct segmentation study due to different number of test images, results indicate that adding either cortical or lacunar infarcts as their own class to the training data increases the WMH segmentation performance. However, segmenting both cortical and lacunar infarcts in addition to the WMH didn't result better WMH or infarct segmentation performance (see Table 5.5). Especially, the number of false positives in detected infarcts increased compared to models in which only cortical or lacunar infarcts are segmented in addition to the WMH. The highest WMH score was obtained using WMH and lacunar infarct segmentation model which was trained with FLAIR, T1 and T2 images. The distributions of false positive and false negative labeled WMH voxels are visualized in Figure 5.3. Comparing FP and FN WMH distributions to the WMH distribution (visualized in Figure 5.1) shows that the majority of the wrongly labeled voxels are located in the areas where WMH is usually detected.

Also, the average dice score over the whole result image set can be a bit misleading for evaluating the performance of the uResNet because the majority of the cases which achieved below average Dice scores are having small WMH volumes. This is visualized in Figure 5.2. When WMH volumes are small, single wrongly labeled voxel has a bigger impact on numerical evaluation resulting lower scores even if visual inspection is showing that quality of the segmentations is good.

Comparison to the other methodologies and studies is not objectively easy because their implementations and data sets are not publicly available. Different MR imaging protocols, data sets and reference images influence results so much that comparing different methods and studies do not provide a fair information between the different methods. However, in order to get general idea how results of this thesis compare with other studies, some results from previous research projects are presented. Guerrero et al. [16] used identical CNN to segment both WMH and cortical infarct lesions from brain images achieving 69.5 average Dice score for WMH lesions and 40.0 for infarct lesions. Ghafoorian et al. [13] presented 0.792 average Dice score using location sensitive deep CNN. Ghafoorian et al. [12] also studied detection of the lacunar infarcts achieving very good 0.974 sensitivity using 3D CNN. However, the number of FP per slice was 0.13. One earlier method proposed by Uchiyama et al. [57] used eigenvector template matching reported to have 0.968 sensitivity and 0.47 FP per slice.

The advantages for proposed uResNet model over traditional image segmentation methods is that once the model is trained, the segmentation process for one image is done in seconds. Also, the model is capable of segmenting multiple structures from brain images at the same time. These advantages and achieved segmentation results suggest that developing an accurate deep learning based commercial segmentation tool is possible.

7. CONCLUSIONS

The aim of this thesis was to develop a segmentation method for white matter hyperintensities and infarcts from brain images using deep learning methods. Work included implementation of image preprocessing pipeline, training and testing the CNN and post-processing the results. Multiple different studies were made in order to find best performing model. For WMH segmentation, differences between 3D and 2D models was studied and this was done using multiple different set of input channels. Infarct detection and segmentation was done in three parts. Firstly, cortical infarcts were segmented alongside with WMH and then lacunar infarcts in addition to the WMH. Finally, both infarcts were segmented in addition to WMH. The numerical evaluation was performed for every set of result images.

The study results show that the designed image analysis system is cabable of segmenting WMH accurately. The best Dice score for WMH volumes was 0.774 and correlation was close to 1. However, segmenting cortical and lacunar infarcts was trickier task. The highest achieved sensitivities were 0.71 for cortical infarcts and only 0.503 for lacunar infarcts. Also, the amount of false positive segmentations was a problem in both cases.

From the results can be deduced that deep learning methods are very interesting and potential way to solve image analysis problems for brain images. The achieved results are promising but further research is still needed in order to improve segmentation results, especially for infarct detection. This study did not focus on fine tuning the CNN and in that area some improvement could be done. Also, the quality of training data and lack of infarcts were the biggest issues for not achieving better performance for detecting cortical and lacunar infarcts. However, the proposed CNN is very powerful tool for segmenting WMH from brain images. Once the model is trained, the actual segmentation process for sample images is done in seconds leading to accurate separation of WMH from other brain tissues.

REFERENCES

[1] R. Adams and L. Bischof, "Seeded region growing," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 16, no. 6, pp. 641–647, 1994.

[2] M. A. Balafar, A. R. Ramli, M. I. Saripan, and S. Mashohor, "Review of brain mri image segmentation methods," *Artificial Intelligence Review*, vol. 33, no. 3, pp. 261–274, 2010.

[3] P. Baldi and S. Brunak, *Bioinformatics: The Machine Learning Approach*, 2nd ed. MIT Press, 2014.

[4] Y. Bengio, I. Goodfellow, and A. Courville, ""deep learning," 2016, book in preparation for mit press," MIT Press, [online],http://www.deeplearningbook.org, 2016.

[5] Z. Cai, C. Wang, W. He, H. Tu, Z. Tang, M. Xiao, and L.-J. Yan, "Cerebral small vessel disease and alzheimer's disease," *Clinical interventions in aging*, vol. 10, p. 1695, 2015.

[6] M. E. Caligiuri, P. Perrotta, A. Augimeri, F. Rocca, A. Quattrone, and A. Cherubini, "Automatic detection of white matter hyperintensities in healthy aging and pathology using magnetic resonance imaging: A review," *Neuroinformatics*, vol. 13, no. 3, pp. 261–276, 2015.

[7] M. J. Cardoso, C. H. Sudre, M. Modat, and S. Ourselin, "Template-based multimodal joint generative model of brain data," in *International Conference on Information Processing in Medical Imaging*. Springer, 2015, pp. 17–29.

[8] Casemed, "Clinical differentiation: Cortical vs. subcortical strokes," http://casemed.case.edu/clerkships/neurology/neurlrngobjectives/neurlrngobj_stroke01new.htm, Case Western Reserve University, School of Medicine, accessed: 2017-12-15.

[9] R. J. Castellani, R. K. Rolston, and M. A. Smith, "Alzheimer disease," *Disease-a-month: DM*, vol. 56, no. 9, p. 484, 2010.

[10] G. Chauhan and S. Debette, "Genetic risk factors for ischemic and hemorrhagic stroke," *Current cardiology reports*, vol. 18, no. 12, p. 124, 2016.

[11] M. Erihov, S. Alpert, P. Kisilev, and S. Hashoul, "A cross saliency approach to asymmetry-based tumor detection," in *International Conference on Medical*

Image Computing and Computer-Assisted Intervention. Springer, 2015, pp. 636–643.

[12] M. Ghafoorian, N. Karssemeijer, T. Heskes, M. Bergkamp, J. Wissink, J. Obels, K. Keizer, F.-E. de Leeuw, B. van Ginneken, E. Marchiori, *et al.*, "Deep multi-scale location-aware 3d convolutional neural networks for automated detection of lacunes of presumed vascular origin," *NeuroImage: Clinical*, vol. 14, pp. 391–399, 2017.

[13] M. Ghafoorian, N. Karssemeijer, T. Heskes, I. W. Uden, C. I. Sanchez, G. Litjens, F.-E. Leeuw, B. Ginneken, E. Marchiori, and B. Platel, "Location sensitive deep convolutional neural networks for segmentation of white matter hyperintensities," *Scientific Reports*, vol. 7, no. 1, p. 5110, 2017.

[14] S. Gollapudi, *Practical Machine Learning*. Packt Publishing, 2016.

[15] L. T. Grinberg and H. Heinsen, "Toward a pathological definition of vascular dementia," *Journal of the neurological sciences*, vol. 299, no. 1, pp. 136–138, 2010.

[16] R. Guerrero, C. Qin, O. Oktay, C. Bowles, L. Chen, R. Joules, R. Wolz, M. Valdes-Hernandez, D. Dickie, J. Wardlaw, *et al.*, "White matter hyperintensity and stroke lesion segmentation and differentiation using convolutional neural networks," *arXiv preprint arXiv:1706.00935*, 2017.

[17] G. Hall, "Pearson's correlation coefficient," *other words*, vol. 1, no. 9, 2015.

[18] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning - Data Mining, Inference, and Prediction*, 2nd ed. Springer, 2009.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *European Conference on Computer Vision*. Springer, 2016, pp. 630–645.

[20] C. Hecht-Leavitt, J. Gomori, R. Grossman, H. Goldberg, D. Hackney, R. Zimmerman, and L. Bilaniuk, "High-field mri of hemorrhagic cortical infarction." *American journal of neuroradiology*, vol. 7, no. 4, pp. 581–585, 1986.

[21] D. L. Hill, P. G. Batchelor, M. Holden, and D. J. Hawkes, "Medical image registration," *Physics in medicine & biology*, vol. 46, no. 3, p. R1, 2001.

[22] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning*, 2015, pp. 448–456.

[23] L. Itti, L. Chang, and T. Ernst, "Segmentation of progressive multifocal leukoencephalopathy lesions in fluid-attenuated inversion recovery magnetic resonance imaging," *Journal of Neuroimaging*, vol. 11, no. 4, pp. 412–417, 2001.

[24] C. R. Jack, P. C. O'Brien, D. W. Rettman, M. M. Shiung, Y. Xu, R. Muthupillai, A. Manduca, R. Avula, and B. J. Erickson, "Flair histogram segmentation for measurement of leukoaraiosis volume," *Journal of Magnetic Resonance Imaging*, vol. 14, no. 6, pp. 668–676, 2001.

[25] M. Jenkinson and S. Smith, "A global optimisation method for robust affine registration of brain images," *Medical image analysis*, vol. 5, no. 2, pp. 143–156, 2001.

[26] J. Juntu, J. Sijbers, D. Van Dyck, and J. Gielen, "Bias field correction for mri images," *Computer Recognition Systems*, pp. 543–551, 2005.

[27] K. Kamnitsas, C. Ledig, V. F. Newcombe, J. P. Simpson, A. D. Kane, D. K. Menon, D. Rueckert, and B. Glocker, "Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation," *Medical image analysis*, vol. 36, pp. 61–78, 2017.

[28] Karpathy. "convolutional neural networks, convolutional neural networks for visual recognition.". Accessed: 2017-12-30. [Online]. Available: http://cs231n.github.io/convolutional-networks/

[29] B. Kayalibay, G. Jensen, and P. van der Smagt, "Cnn-based segmentation of medical imaging data," *arXiv preprint arXiv:1701.03056*, 2017.

[30] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[31] J. Koikkalainen, H. Rhodius-Meester, A. Tolonen, F. Barkhof, B. Tijms, A. W. Lemstra, T. Tong, R. Guerrero, A. Schuh, C. Ledig, *et al.*, "Differential diagnosis of neurodegenerative diseases using structural mri data," *NeuroImage: Clinical*, vol. 11, pp. 435–449, 2016.

[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[33] C. Laurent, G. Pereyra, P. Brakel, Y. Zhang, and Y. Bengio, "Batch normalized recurrent neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on.* IEEE, 2016, pp. 2657–2661.

[34] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.

[35] L. K. Lee, S. C. Liew, and W. J. Thong, "A review of image segmentation methodologies in medical image," in *Advanced computer and communication engineering technology*. Springer, 2015, pp. 1069–1080.

[36] Q. Li, W. Cai, X. Wang, Y. Zhou, D. D. Feng, and M. Chen, "Medical image classification with convolutional neural network," in *Control Automation Robotics & Vision (ICARCV), 2014 13th International Conference on*. IEEE, 2014, pp. 844–848.

[37] G. Litjens, T. Kooi, B. Bejnordi, A. Setio, F. Ciompi, M. Ghafoorian, J. van der Laak, B. van Ginneken, and C. Sánchez, "A survey on deep learning in medical image analysis. arxiv preprint (2017)," *arXiv preprint arXiv:1702.05747*.

[38] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[39] J. M. Lötjönen, R. Wolz, J. R. Koikkalainen, L. Thurfjell, G. Waldemar, H. Soininen, D. Rueckert, A. D. N. Initiative, *et al.*, "Fast and robust multi-atlas segmentation of brain magnetic resonance images," *Neuroimage*, vol. 49, no. 3, pp. 2352–2365, 2010.

[40] MILA. Theano. University of Montreal. Accessed: 2018-02-01. [Online]. Available: http://deeplearning.net/software/theano/

[41] Nvidia. cudnn. Accessed: 2018-02-01. [Online]. Available: https://developer.nvidia.com/cudnn

[42] F. P. Oliveira and J. M. R. Tavares, "Medical image registration: a review," *Computer methods in biomechanics and biomedical engineering*, vol. 17, no. 2, pp. 73–93, 2014.

[43] A. Poggesi, A. Gouw, W. van der Flier, G. Pracucci, H. Chabriat, T. Erkinjuntti, F. Fazekas, J. M. Ferro, C. Blahak, P. Langhorne, *et al.*, "Neurological abnormalities predict disability: the ladis (leukoaraiosis and disability) study," *Journal of neurology*, vol. 261, no. 6, pp. 1160–1169, 2014.

[44] C. Qin, R. G. Moreno, C. Bowles, C. Ledig, P. Scheltens, F. Barkhof, H. Rhodius-Meester, B. Tijms, A. W. Lemstra, W. M. van der Flier, *et al.*,

"A semi-supervised large margin algorithm for white matter hyperintensity segmentation," in *International Workshop on Machine Learning in Medical Imaging*. Springer, 2016, pp. 104–112.

[45] J. Ramirez, A. A. McNeely, C. J. Scott, M. Masellis, S. E. Black, A. D. N. Initiative, *et al.*, "White matter hyperintensity burden in elderly cohort studies: The sunnybrook dementia study, alzheimer's disease neuroimaging initiative, and three-city study," *Alzheimer's & Dementia*, vol. 12, no. 2, pp. 203–210, 2016.

[46] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 234–241.

[47] À. Rovira and A. León, "Mr in the diagnosis and monitoring of multiple sclerosis: an overview," *European journal of radiology*, vol. 67, no. 3, pp. 409–414, 2008.

[48] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[49] P. Scheltens, "Imaging in alzheimer's disease," *Dialogues in clinical neuroscience*, vol. 11, no. 2, p. 191, 2009.

[50] F. B. P. Scheltensb, "Imaging of white matter lesions," *Cerebrovasc Dis*, vol. 13, no. 2, pp. 21–30, 2002.

[51] D. Scherer, A. Müller, and S. Behnke, "Evaluation of pooling operations in convolutional architectures for object recognition," *Artificial Neural Networks–ICANN 2010*, pp. 92–101, 2010.

[52] P. Schmidt. A lesion segmentation tool for spm. Accessed: 2017-12-15. [Online]. Available: http://www.statistical-modelling.de/lst.html

[53] D. Shen, G. Wu, and H.-I. Suk, "Deep learning in medical image analysis," *Annual Review of Biomedical Engineering*, no. 0, 2017.

[54] E. Smistad, T. L. Falch, M. Bozorgi, A. C. Elster, and F. Lindseth, "Medical image segmentation on gpus–a comprehensive review," *Medical image analysis*, vol. 20, no. 1, pp. 1–18, 2015.

[55] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting." *Journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[56] N. J. Tustison, B. B. Avants, P. A. Cook, Y. Zheng, A. Egan, P. A. Yushkevich, and J. C. Gee, "N4itk: improved n3 bias correction," *IEEE transactions on medical imaging*, vol. 29, no. 6, pp. 1310–1320, 2010.

[57] Y. Uchiyama, A. Abe, C. Muramatsu, T. Hara, J. Shiraishi, and H. Fujita, "Eigenspace template matching for detection of lacunar infarcts on mr images," *Journal of digital imaging*, vol. 28, no. 1, pp. 116–122, 2015.

[58] T. van Laarhoven, "L2 regularization versus batch and weight normalization," *arXiv preprint arXiv:1706.05350*, 2017.

[59] E. C. Van Straaten, F. Fazekas, E. Rostrup, P. Scheltens, R. Schmidt, L. Pantoni, D. Inzitari, G. Waldemar, T. Erkinjuntti, R. Mäntylä, *et al.*, "Impact of white matter hyperintensities scoring method on correlations with clinical data: the ladis study," *Stroke*, vol. 37, no. 3, pp. 836–840, 2006.

[60] J. Wang and L. Perez, "The effectiveness of data augmentation in image classification using deep learning," Technical report, Tech. Rep., 2017.

[61] Y. Wang, J. A. Catindig, S. Hilal, H. W. Soon, E. Ting, T. Y. Wong, N. Venketasubramanian, C. Chen, and A. Qiu, "Multi-stage segmentation of white matter hyperintensity, cortical and lacunar infarcts," *Neuroimage*, vol. 60, no. 4, pp. 2379–2388, 2012.

[62] J. M. Wardlaw, M. C. V. Hernández, and S. Muñoz-Maniega, "What are white matter hyperintensities made of? relevance to vascular cognitive impairment," *Journal of the American Heart Association*, vol. 4, no. 6, p. e001140, 2015.

[63] W. Wen and P. Sachdev, "The topography of white matter hyperintensities on brain mri in healthy 60-to 64-year-old individuals," *Neuroimage*, vol. 22, no. 1, pp. 144–154, 2004.

[64] WHO. Dementia. Accessed: 2017-12-15. [Online]. Available: http://www.who.int/mediacentre/factsheets/fs362/en/

[65] D. H. Ye, D. Zikic, B. Glocker, A. Criminisi, and E. Konukoglu, "Modality propagation: coherent synthesis of subject-specific scans with data-driven regularization," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2013, pp. 606–613.

[66] O. Zanetti, S. Solerte, and F. Cantoni, "Life expectancy in alzheimer's disease (ad)," *Archives of gerontology and geriatrics*, vol. 49, pp. 237–243, 2009.

[67] B. Zitova and J. Flusser, "Image registration methods: a survey," *Image and vision computing*, vol. 21, no. 11, pp. 977–1000, 2003.